

# Open Research Online

---

The Open University's repository of research publications and other research outputs

## Legal knowledge engineering: Computing, logic and law

### Thesis

#### How to cite:

Leith, Philip (1986). Legal knowledge engineering: Computing, logic and law. PhD thesis The Open University.

For guidance on citations see [FAQs](#).

© 1985 The Author



<https://creativecommons.org/licenses/by-nc-nd/4.0/>

Version: Version of Record

Link(s) to article on publisher's website:

<http://dx.doi.org/doi:10.21954/ou.ro.0000de52>

---

Copyright and Moral Rights for the articles on this site are retained by the individual authors and/or other copyright owners. For more information on Open Research Online's data [policy](#) on reuse of materials please consult the policies page.

---

[oro.open.ac.uk](http://oro.open.ac.uk)

D 68519/86

UNRESTRICTED

LEGAL KNOWLEDGE ENGINEERING:

Computing, Logic and Law

Philip Leith

B.Sc., Dip(SAD)

A thesis submitted for the degree of

Doctor of Philosophy

Computing Discipline

Open University

England

Author's Number: HDL 5467

Date of Submission: January 1985

Date of Award: 20.3.86

14TH JANUARY 1985.

**APPENDIX PAGES 265-269**

**NOT DIGITISED BY REQUEST OF THE  
UNIVERSITY**

## Published Work

Sections of this thesis have appeared as various published papers, either in the form as contained here or in relatively early and undeveloped form. These papers are not referred to directly within the text, but are:

"ELI: An expert legislative consultant", in Proceedings IEE Conference on Man/Machine Systems, Manchester, 1982.

"Hierarchically Structured Production Rules", in Computer Journal, Vol. 26, No. 1, 1983.

"Cautionary Notes on Legal Expert Systems", in Computers and Law, No. 40, May 1984.

"Logic, Formal Models and Legal Reasoning", in Jurimetrics, Vol. 24, No. 4, Summer 1984.

"Logic and the Law", Alvey Workshop on Inference, Imperial College, September 1984.

"An IKBS Implementation", in Software Practice and Experience, Vol 15(1), January 1985.

"Law and Computer Program: The Limits to Logic", presented at the 12th World Congress on Philosophy of Law and Social Philosophy, Athens, August 1985.

"Clear Rules and Legal Expert Systems", presented at "Logica, Informatica, Diritto", Florence, September 1985.

"Legal Expertise and Legal Expert Systems", to appear in Yearbook of Law, Computers, Technology, Butterworth's, Vol 2, 1986.

"Fundamental Errors in Legal Logic Programming", to appear in Computer Journal, 1986.



### Acknowledgements

My thanks to the Deans of the Faculty of Mathematics at The Open University for encouraging my research by making time and resources available to me; to Professor R.J. Housden for positively continuing this encouragement; to Dr. D.C. Ince for various forms of supervision; to Professor D.N. MacCormick for comments upon Chapter 4, and its earlier versions; and to Dr. B.P. Bloomfield and Dr. R.N. Moles for interdisciplinary discussions on, for instance, neo-Platonism and the existence of rules.

## The Problem

"(A lawyer robot) with a far grater memory capacity than any of the others and with a brain-computer that operates on logic. That's what the law is, isn't it - logic?"

"I suppose it is," said Lee. "At least its supposed to be ... It just wouldn't work. To practice law, you must be admitted to the bar. To be admitted to the bar, you must have a degree in law and pass an examination and, although there's never been an occasion to establish a precedent, I suspect the applicant must be human." ...

"All they'd need to do would be read the books," said Albert. "Ten seconds to a page or so. Everything they read would be stored in their memory cells." ...

Lee scrubbed his chin with a knotted fist and the light of speculation was growing in his eyes. "It might be worth a try. If it works, though, it'll be an evil day for jurisprudence."

Clifford D. Simak

Quoted by [Bing, p61, 1977]

The Open University  
Higher Degrees Office  
- 4 JUN 1986  
EX35  
ACK .....  
PASS TO .....  
DISPOSAL .....

HIGHER DEGREES OFFICE

LIBRARY AUTHORISATION

STUDENT: ... PHILIP LEITCH ..... SERIAL NO: .....

DEGREE: ... PHD .....

TITLE OF THESIS: "LEGAL KNOWLEDGE ENGINEERING  
COMPUTING, LOGIC AND LAW"  
.....  
.....

I confirm that I am willing that my thesis be made available to readers  
and may be photocopied, subject to the discretion of the Librarian.

Signed: ..... Date: ..... / ..... / .....

## Abstract

The general problem approached in this thesis is that of building computer based legal advisory programs (otherwise known as expert systems or Intelligent Knowledge Based Systems). Such computer systems should be able to provide an individual with advice about either the general legal area being investigated, or advice about how the individual should proceed in a given case.

In part the thesis describes a program (the ELI program) which attempts to confront some of the problems inherent in the building of these systems. The ELI system is seen as an experimental program (currently handling welfare rights legislation) and development vehicle. It is not presented as a final commercially implementable program. We present a detailed criticism of the type of legal knowledge contained within the system.

The second, though in part intertwined, major subject of the thesis describes the jurisprudential aspects of the attempt to model the law by logic, a conjunction which is seen to be at the heart of the computer/law problem. We suggest that the conjunction offers very little to those who are interested in the real application of the real law, and that this is most forcefully seen when a working computer system models that conjunction.

Our conclusion is that neither logic nor rule-based methods are sufficient for handling legal knowledge. The novelty and import of this thesis is not simply that it presents a negative conclusion; rather that it offers a sound theoretical and pragmatic framework for understanding why these methods are insufficient - the limits to the field are, in fact, defined.

# CONTENTS

|           |  |     |
|-----------|--|-----|
| CHAPTER 1 | INTRODUCTION   |     |
| 1.1       | OVERVIEW . . . . .   | 1   |
| 1.2       | THE ELI PROGRAM . . . . .                                    | 2   |
| 1.3       | EXTENSIONS AND CRITIQUE . . . . .                            | 5   |
| CHAPTER 2 | COMPUTING, LOGIC AND LAW                                     |     |
| 2.1       | INTRODUCTION . . . . .                                       | 7   |
| 2.2       | LOGIC AND THE LAW . . . . .                                  | 7   |
| 2.3       | LEGAL COMPUTING . . . . .                                    | 12  |
| CHAPTER 3 | THE KNOWLEDGE BASED FRAMEWORK                                |     |
| 3.1       | LEGAL KNOWLEDGE ENGINEERING . . . . .                        | 17  |
| 3.2       | INTELLIGENT KNOWLEDGE BASED SYSTEMS . . . . .                | 22  |
| 3.3       | PERCEIVED ADVANTAGES OF IKBS TECHNIQUES . . . . .            | 26  |
| 3.4       | ARTIFICIAL INTELLIGENCE, LEARNING AND KNOWLEDGE . . . . .    | 29  |
| CHAPTER 4 | LOGIC, FORMAL MODELS AND LEGAL REASONING                     |     |
| 4.1       | INTRODUCTION . . . . .                                       | 36  |
| 4.2       | THE NATURE AND ROLE OF LOGIC. . . . .                        | 37  |
| 4.3       | EVERYDAY LOGIC . . . . .                                     | 38  |
| 4.4       | LOGIC AND TRUTH . . . . .                                    | 40  |
| 4.5       | CONSISTENCY OF LOGICS . . . . .                              | 47  |
| 4.6       | THE NOTION OF CLASSES . . . . .                              | 48  |
| 4.7       | APPLICATION OF LOGIC TO THE LAW . . . . .                    | 51  |
| 4.7.1     | Previous Applications . . . . .                              | 51  |
| 4.7.2     | A Logical Jurisprudential Model . . . . .                    | 54  |
| 4.8       | A JURISPRUDENTIAL MODEL OF LEGAL REASONING . . . . .         | 55  |
| 4.8.1     | Deduction . . . . .  | 55  |
| 4.8.2     | Production Systems . . . . .                                 | 56  |
| 4.8.3     | A Legal Example . . . . .                                    | 60  |
| 4.9       | THE AFFINITY OF THE MODEL TO ACTUAL LEGAL REASONING. . . . . | 67  |
| 4.9.1     | Introduction. . . . .  | 67  |
| 4.9.2     | Psychologicalism . . . . .                                   | 67  |
| 4.9.3     | Deduction Or Induction . . . . .                             | 69  |
| 4.9.4     | Law As A Formal System . . . . .                             | 71  |
| 4.9.5     | Conclusion: The Myth Of Logical Compulsion . . . . .         | 73  |
| CHAPTER 5 | A SHORT ANALYSIS OF CURRENT LEGAL CONSULTATIVE SYSTEMS       |     |
| 5.1       | INTRODUCTION . . . . .                                       | 87  |
| 5.2       | THE TAXMAN PROJECT . . . . .                                 | 89  |
| 5.3       | THE MELDMAN PROJECT . . . . .                                | 98  |
| 5.4       | THE LEGOL PROJECT . . . . .                                  | 100 |
| 5.5       | THE INVERCLYDE PROJECT . . . . .                             | 102 |
| 5.6       | THE PROLOG PROJECT . . . . .                                 | 105 |
| CHAPTER 6 | THE ELI PROJECT  |     |
| 6.1       | INTRODUCTION . . . . .                                       | 114 |

|         |  |     |
|---------|--|-----|
| 6.2     | RULES WITHIN THE ELI PROGRAM . . . . . | 116 |
| 6.2.1   | Rule Content . . . . .                 | 116 |
| 6.2.2   | Definitional Rules . . . . .           | 121 |
| 6.2.3   | Handling Negation . . . . .            | 123 |
| 6.2.3.1 | Pruning The Knowledge Base . . . . .   | 125 |
| 6.2.4   | Control . . . . .                      | 127 |
| 6.3     | CONCEPTUAL MATCHING . . . . .          | 131 |

## CHAPTER 7 ELI IMPLEMENTATION

|         |  |     |
|---------|--|-----|
| 7.1     | INTRODUCTION . . . . .                         | 134 |
| 7.1.1   | Rules And Knowledge Bases . . . . .            | 135 |
| 7.1.2   | Structuring The Rules In A Graph. . . . .      | 137 |
| 7.1.3   | Incorporating Rules . . . . .                  | 139 |
| 7.2     | THE PROGRAMMED SYSTEM . . . . .                | 140 |
| 7.2.1   | Overview Of System Parts . . . . .             | 140 |
| 7.2.2   | Implementation Of The Knowledge Base . . . . . | 141 |
| 7.2.2.1 | Handling Conditions . . . . .                  | 141 |
| 7.2.2.2 | Handling Goals . . . . .                       | 145 |
| 7.2.2.3 | Addressing . . . . .                           | 145 |
| 7.2.2.4 | Creating Planes And Goals . . . . .            | 147 |
| 7.2.2.5 | Structure Traversal . . . . .                  | 148 |
| 7.2.3   | Interpretation . . . . .                       | 150 |
| 7.2.3.1 | Decomposition . . . . .                        | 150 |
| 7.2.3.2 | Negation . . . . .                             | 154 |
| 7.2.4   | Conceptual Matching . . . . .                  | 156 |
| 7.2.5   | Building The Knowledge Base . . . . .          | 158 |
| 7.2.6   | Annotating The Knowledge Base . . . . .        | 160 |
| 7.2.7   | System Tools . . . . .                         | 161 |
| 7.3     | A NOTE ON LANGUAGE . . . . .                   | 162 |

## CHAPTER 8 FURTHER RESEARCH AND DEVELOPMENT WITH RULE-BASED TECHNIQUES.

|         |   |     |
|---------|---|-----|
| 8.1     | INTRODUCTION . . . . .                                  | 165 |
| 8.2     | SHORT TERM DEVELOPMENT OF THE ELI PROGRAM . . . . .     | 165 |
| 8.2.1   | Extending Meta-Level Control . . . . .                  | 166 |
| 8.2.2   | Query By Assertion And Explanation Facilities . . . . . | 170 |
| 8.2.3   | Utilizing An Algorithmic Language . . . . .             | 171 |
| 8.2.4   | Revise The SETUP Facility . . . . .                     | 172 |
| 8.2.5   | Conclusion . . . . .                                    | 173 |
| 8.3     | MAN/MACHINE CONSIDERATIONS . . . . .                    | 173 |
| 8.3.1   | Lawyer/machine Interaction . . . . .                    | 173 |
| 8.3.1.1 | Legal Problems . . . . .                                | 175 |
| 8.3.1.2 | Persuading Potential Users . . . . .                    | 180 |

## CHAPTER 9 THE LIMITS TO RULES AND LOGIC IN LEGAL KNOWLEDGE ENGINEERING

|       |   |     |
|-------|---|-----|
| 9.1   | INTRODUCTION . . . . .  | 185 |
| 9.2   | CLEAR RULES . . . . .   | 189 |
| 9.3   | CLEAR RULES IN ELI . . . . .  | 193 |
| 9.4   | A CRITIQUE OF CLEAR RULES . . . . .                                 | 195 |
| 9.5   | CLEAR RULES AND LOGIC IN LEGAL EXPERT SYSTEMS . . . . .             | 205 |
| 9.5.1 | Negotiation Not Clarity . . . . .                                   | 205 |
| 9.5.2 | The Ouster Clause - A Problem For Formalization<br>Of Law . . . . . | 208 |
| 9.5.3 | The Ouster Clause . . . . .   | 210 |
| 9.6   | CONSERVATIVELY APPLYING THE RULES: LOGIC AGAIN . . . . .            | 216 |
| 9.7   | SYNOPSIS OF THE CHALLENGE TO LEGAL LOGICIANS . . . . .              | 221 |

|            |   |     |
|------------|---|-----|
| CHAPTER 10 | CONCLUSION: AFTER RULE-BASED SYSTEMS?         |     |
| 10.1       | RESEARCH GOALS . . . . .                      | 224 |
| 10.2       | DETERMINISM AND NON-DETERMINISM . . . . .     | 226 |
| APPENDIX A | THE ELI COMMANDS                              |     |
| A.1        | INTRODUCTION . . . . .                        | 231 |
| A.2        | THE SETUP MODULE . . . . .                    | 232 |
| A.3        | THE BUILD MODULE . . . . .                    | 238 |
| A.4        | THE INTERPRET MODULE . . . . .                | 244 |
| APPENDIX B | A FRAGMENT OF ELI IN USE                      |     |
| APPENDIX C | ANOTHER EXAMPLE - EUROPEAN COMMUNITY LAW      |     |
| C.1        | INTRODUCTION . . . . .                        | 252 |
| C.2        | DISCUSSION . . . . .                          | 256 |
| C.2.1      | Conceptual Matching . . . . .                 | 257 |
| C.2.2      | The Use Of Negation . . . . .                 | 258 |
| C.2.3      | Legislative Self-reference . . . . .          | 261 |
| C.2.4      | Difficulties Found In Normalization . . . . . | 262 |
| C.2.5      | Presentations Of Conclusions . . . . .        | 264 |
| APPENDIX D | A LISTING OF PART OF THE ELI KNOWLEDGE BASE   |     |
| APPENDIX E | REFERENCES                                    |     |

## CHAPTER 1

### INTRODUCTION

#### 1.1 OVERVIEW

This thesis describes both the philosophical and pragmatic problems inherent in an attempt to engineer a legal advisory system. Much of the thesis therefore deals with aspects of a jurisprudential nature: also other already existing systems are examined and analyzed. The particular jurisprudential subject is that of logic and the law, because - we argue - such a conjunction is at the very heart of the field of computers and law. We shall argue that there has been much confusion in the research field: confusion caused principally by a misunderstanding of the nature of logic. There is also confusion because current jurisprudence has offered us an unsound epistemology. This unsoundness is particularly relevant to those who wish to build rule-based IKBS systems since it has tended to imply that we are "bound by legal rules".

Why was it not possible to concentrate only upon the computing? Why not simply theorise without coding? There are two reasons. First, it is not enough to simply describe a "planned" program, because in the discipline of computer science no matter how theoretical we might wish to be, the results of our theories must always appear in the form of a coded program. It is a case of, "the proof of the pudding ...". Therefore, in parallel with the jurisprudential research, effort has gone into the production of a program - the ELI program - to both test



ideas and develop techniques and methods for the representing of legal knowledge.

Second, we should point out that it was upon examination of the finished program that our disappointment with the rule-based approach came to light; without the coding - "eating the pudding" - this disappointment might not have occurred. The computer system, therefore, acted as a model of a theory - a theory which when found wanting could be understood only within the differing jurisprudential framework.

It is perhaps beyond the scope of this thesis to suggest that this inter-reaction and inter-utility of computer science and jurisprudence is further useful, since some might believe that the two disciplines are unconnected. I would argue, to the contrary, that many of the problems of computing (for example, the attempts to find artificial intelligence or to prove programs correct) have been mirrored in the philosophy of law (for example, formalism in law or the attempt to fully define legislation in logical terms) - each subject does not have its own epistemology: a proper understanding of knowledge about the world is a need which is common to both.

However, we leave these discussions to elsewhere, and simply give an outline of the ELI program. We shall be relatively uncritical of the program until the final sections of the thesis; then, no holds will be barred.

## 1.2 THE ELI PROGRAM

Briefly, the program utilizes IKBS techniques to handle welfare rights legislation; that is, it operates in Mehl's [Mehl, 1958] consultation mode. The program is seen as a test bed which can usefully be used to handle a variety of areas of the law.

The program accepts input from an expert in the form of if-then rules where the antecedent contains one or more conditions which must be true before the consequent can be proven (for an introduction to the use of these rules see [Waterman, 1978b]). These rules were extracted from relevant Acts of Parliament (that is those dealing with Supplementary Benefit, Housing Benefit etc.) and guides to those Acts (for example [Allbeson, 1984]). This process of extracting rules from legislation is termed "normalization", and requires a high degree of legal skill in most domains. The lack of this legal skill on the author's part (MacCormick in a private note referred to some of our normalization as "dodgy" which points to the fact that interpretation of legislation is encouraged by, if not requiring, legal training) is one reason why the program does not attempt to handle other more complex areas.

As an example of a program rule, the following is one of the rules which define how a client (that is the person who is to be the recipient of the advice) can be eligible for Supplementary Benefit:

IF (applicant is available for work)

(applicant is a British Citizen)

THEN

(applicant satisfies the rules of entitlement to  
Supplementary Benefit)

This can be read as:

If the client is available for work and is a British Citizen, then  
the client can apply for Supplementary Benefit.

There are several other rules which have the same consequent. One of these rules is:

IF (applicant is available for work)

(applicant has been granted foreign husband status,

refugee status or political asylum)

THEN

(applicant satisfies the rules of entitlement to  
Supplementary Benefit)

These rules only designate the basic right of application to Supplementary Benefit - further rules determine whether the client is actually able to receive assistance from the Department of Health and Social Security. The program, among its rules, also includes rules which determine access to and, if relevant, amount of housing benefit and other welfare benefits (e.g. free health, transport to hospitals etc). One aspect of ELI is of importance to both the manner of the program's execution and the easy means of rule input, is the use of negation of individual rule elements and entire rules.

The rules are in a format which can be best described as "legal language"; there are no atomic "legal objects" or atomic "legal relationships" used in the program (such as "study-status-OK" or "is-owned-by"). We discuss the advantages of this later in detail, but would argue that the principal advantage is of ease of use to the legal expert inputting rules from other legislative areas, and that there are areas of legal knowledge which are not amenable to the atomic approach anyway.

Further to this ease, the program is robust enough to attempt to recognize "conceptual matching" between input conditions and goals. This is important where large numbers of rules are being input into the program; in such a case the program will attempt to resolve spelling mistakes and slightly varying wording - "applicant is unemployed" and "client is not in employment" etc. This information then remains with the program so that when future rules are being input, such matches are noticed.

A large number of other tools are available to the person who inputs the rules - searching, deletion, etc.

The architecture of the program is interesting, using a complex data structure to hold these rules. The advantages of this structure arise mainly because the rule elements rather than the rule is the primary element in the knowledge base. The rules can inhibit each other's firing - a form of "meta-level" knowledge - and, self organize themselves in this structure.

The ELI program is a large INTERLISP program which occupies some 90 pages of storage on a DEC-20 mainframe. The necessary "legal" rules (held in the knowledge base) which provide advice covering various Acts of Parliament and case precedent occupy 70 pages of storage.

### 1.3 EXTENSIONS AND CRITIQUE

The program should be seen as a 'test bed' for ideas and methods; we do not see its present form as by any means final, and we have not tested the program in situ (apart from that testing by Helene Bauer-Bernet discussed in the Appendix) since - we are certain - that the current ELI program is not the final system which we would wish. In fact, later in this text we try to analyze how and if it might be practically extended in terms of both the legal area handled and also in terms of incorporating other non rule-based information.

The computer scientist who looks for timing benchmarks and discussion of efficiency will be disappointed with this thesis: it does not contain them. Our justification is quite simple - computational aspects are important when we are near implementing a system in-situ, but thinking about those matters in legal IKBS research is rather like the researcher described in [Catt, 1973] who tried to model the pattern of the brain synapses and suchlike in a computer; that, we would suggest, is task enough without that researcher's concurrent (but also unsuccessful)

attempt to model this system within a head shaped piece of hardware (it, of course, kept overheating). First, catch your intelligence, we might say, then concern yourself with making it computationally efficient (or head shaped).

## CHAPTER 2

### COMPUTING, LOGIC AND LAW

#### 2.1 INTRODUCTION

We here briefly glance at the general and historical position of the conjunction of the disciplines of computing, logic and law; such a brief glance will give a general impression of the questions which each area has taken as its domain of study. Such a view is necessary as a starting point for understanding this thesis.

#### 2.2 LOGIC AND THE LAW

If Ramist logic (from the frenchman, Pierre Ramus) is accepted as a useful logic (which many would not accept - for example, [Ong, 1958]), then perhaps Abraham Fraunce was the earliest proponent of the link between logic and the law:

"If Lawes by reason framed were, and grounded on the same;  
If Logike also reason bee, and thereof had this name;  
I see no reason, why that Law and Logike should not bee  
The nearest and the dearest freends, and therefore best agree."

[Fraunce, 1588]

Fraunce's research (although 'spleen' might be a more appropriate term) was directed against the medieval scholastic interpretation of Aristotelian logic and utilized the Ramist view of logic as a means of explicating what he saw as the dichotomous nature of the law (a precursor, perhaps, to structural anthropology as in [Levi-Strauss, 1966]). His thesis was that the manner in which the law operates is intrinsically logical:

"... I then perceaved, the practise of Law to bee the use of Logike, and the methode of Logike to lighten the Lawe. So that after application of Logike to Lawe, and examination of Lawe by Logike I made playne the precepts of the one by the practice of the other, and called my booke, The Lawyers Logike ... "

[Fraunce, 1588]

It must be said, though, that Ramist logic is little studied now, and perhaps - apart from its anti-scholastic temperament which as Styazhkin [Styazhkin, p50, 1969] notes, influenced Leibniz - is treated as an aberration in the progression of Aristotelian and Stoic logic. Ong most certainly considers it so:

"Ramus was not a great intellectual but a savant with wide-ranging interests whose most distinctive attitudes were superficially revolutionary but at root highly derivative. His way of attacking the genuine weaknesses of the scholastic heritage while preserving unwittingly the basic presuppositions responsible for these weaknesses (and for much strength) made his views congenial to the vast numbers of impatient but not too profound thinkers who became his followers, and it gives both him and them tremendous historical value today."

[Ong, 1958]

While Fraunce was a lawyer with an interest in logic, a logician with an interest in the law was George Boole. Boole, in "The Laws of Thought" used a legal example in much the same way that Fraunce had analyzed the case of Lord Northumberland. Boole though, instead of analyzing a specific case, set out in logical symbolism part of the Jewish dietary laws:

"Here, as before, x standing for "clean beasts", y for "beasts dividing the hoof", z for "beasts chewing the cud," we have

$$x = yx;$$

whence

$$1 - x = 1 - yz;$$

and developing the second member,

$$1 - x = y(1-z)+z(1-y)+(1-y)(1-z)$$

which is interpreted into the following Proposition: Unclean beasts are all which divide the hoof without chewing the cud, all which chew the cud without dividing the hoof, and all which neither divide the hoof nor chew the cud."

[Boole, p93/4, 1854]

This interest in the connection between logic and the law has grown since Boole's logical work became generally available. One reason for this is simply because of the utility that such a connection might have. Logic is seen to be well formalized, accessible to those who might shy away from more advanced mathematical theory and has proven its capacity in much of mathematics (one mathematician has informed me - in conversation - that he could not envisage the discipline without its set theoretic aspect, though its introduction is relatively recent); if these advantages could be carried over to the study of law, then the



lawyer, jurisprudentialist and client might all benefit. Notice that this argument of mathematical formality's utility is similar to that used by computer scientists interested in the formal specification of programs, for example, Milne and Strachey [Milne, 1976].

It is also the case that logic has been seen as the realization of "correct reasoning". As MacCormick, for example, has said:

"we are concerned not with the demonstration of logical truths but with their application, that is with the application of logically valid forms of argument in legal contexts. It is important therefore to emphasize that the logical validity of an argument does not guarantee the truth of its conclusion; that the argument is valid entails that if the premisses are true, the conclusion must be true; but logic itself cannot establish or guarantee the truth of the premisses."

[MacCormick, p25, 1978]

and also from computer science:

"The principal application of formal methods lies in the area of verification, where mathematical techniques are used to argue that a given program is correct."

[Berg, 1982]

The interest in this legal/logical connection is still high - perhaps one contemporary reason is that it easily integrates itself with the normative schools of Jurisprudence (i.e. those who see the law as being rule based, for example Hart [Hart, 1961]); these legal norms conveniently act as propositions of Modus Ponens. MacCormick [MacCormick, 1978] is one who currently argues the importance of seeing the law, or at least one part of the legal process (the judicial justification), as following the method of Modus Ponens.

Apart from MacCormick's analytical approach there are many prescriptive researches taking place (by which we mean that the researcher is attempting to change the legal method, as opposed to analyzing it). One of them is directed towards the very difficult problem of legal drafting (see for example the Renton Report [Renton, 1975]) - a problem which has much in common with the definition of programming languages (see for instance, [McGettrick, 1980]). Layman E. Allen has argued that the use of logical drafting techniques may overcome many of the purely semantic difficulties of writing legislation in a non-ambiguous way:

"This imprecision can be categorized into two types of uncertainty: the uncertainty that results from what is omitted from the writing and the uncertainty that results from what is written. Both of these types of uncertainty may be deliberate by the drafter of the document, or they may occur inadvertently. Here, attention is being focused on the inadvertent uncertainties."

[Allen, p75, 1980]

To do this, a logical formalism might be incorporated into the drafting process (thus, he claims, logical operators - AND, OR - would add a non-ambiguous structure to legislation). Note, though, that Allen does not claim that "law should become logic". It is interesting to note that Allen has also shown that not only the normative view of legislation can integrate logic into its theoretical position, but also that view of law as concept directed, for example Hohfeldian Analysis [Allen, 1974]. There are some criticisms of this use of logic in the law, though, Summers for example [Summers, 1961]. We add to this criticism later.

We look at some of these aspects in more detail in Chapter 4, but some workers in the field are: Del Vecchio [Del Vecchio, 1914] with an interest in 'the formal bases of law', Guest [Guest, 1961] who usefully draws attention to the variety of logical systems and Tammelo [Tammelo,

1969] who utilizes Polish logic as a description of the law. Another approach is de Mulder's et al "Sentencing by Computer" which approaches the subject from a slightly different tack [Mulder, 1982].

There is a divide between those who are interested in "law and logic" which might be distinguished as a Platonic/non Platonic divide. The view that logic is intrinsic to "the law" might be considered Platonic. We take the opposing view and would claim that logic can only model the law, never describe it. Such is the perspective taken in Chapter 4 where we describe our relativist position on logic.

The existence of a machine which can represent logical processes has, as might be expected, done much to increase research interest in logic and the law. This has sometimes led to there being a very fine dividing line between research into logic and the law and research into legal computing. The LEGOL project is an example of this overlapping of research borders where the aims of the project have been stated as:

"The ultimate goal of this research is to specify a formal language, supported by a computer system to interpret it, capable of expressing rules of organizations. The rules we are studying are those in legislation. If LEGOL can be developed to handle legal norms then the range of applications will be very wide indeed."

[LEGOL project handout, undated]

Next we examine some research which has arisen from the computer science field.

## 2.3 LEGAL COMPUTING

Precursors to the idea that electronic computers could be used to aid the legal process have been pointed out by Bing and Harvold:

"The first modern electronic computer - ENIAC - was operational in the summer of 1946. That same year Lewis O. Kelso pointed out:

'Today the lawyer works substantially as he worked before the industrial revolution. Only automated legal research will save him from playing one of the most confused, ill-paid and unsatisfactory professions in the world of tomorrow.'

Kelso's proposal is believed to be the earliest suggestion for creating automatic retrieval systems to assist legal research. His suggestion was stimulated by the work of Dr. Vannevar Bush, who had advocated mechanical searching methods in scientific fields, and Kelso suggested a "Law-dex"-system based on the use of punched cards.

Another suggestion was made in 1955 by Vincent P. Biunno of the New Jersey Law Institute. He proposed to enter legal information on a tape which was to be moving continuously past a number of read-out stations. The information might then be retrieved by different lawyers more or less at the same time."

[Bing, p60, 1977].

The first researcher to actually propose the use of a computer was Lucien Mehl [Mehl, 1958]. While Mehl continued to describe what we might now think of as quaint computing techniques (for example, translation of legal arguments directly into binary format) he did make a fundamental divide between two types of "law machine". The first type he described as the "documentary machine" and the second the "consultation machine":

"One may imagine two basic types of law machine:

(1) the documentary or information machine, or - in more familiar terms - the machine for finding the precedent (or relevant text),

(2) the consultation machine; less properly, the "judgement machine".

There is no fundamental difference between these two types of machine; the difference is one of degree rather than essence. The consultation machine will give an exact answer to any put to it, whereas the information machine will only supply a set of items of information bearing on the problem. Conceptual and relational analysis is more acute in the consultation machine; its structure is more delicate, the network of information is more finely woven."

[Mehl, p759, 1958]

It is proposed in this thesis that there is, in fact, a fundamental difference between the two types of law machine; that, however, should not detract from the importance of Mehl's other insights.

The computerized legal information retrieval system has developed quickly and successfully since Mehl's paper - for example, the LEXIS system [Rubin, 1976] and JURIS [Kondos, 1974] system are now available to most legal researchers. The history of the development of these systems is interesting because of the lessons it might offer to the consultation type machines.

The development of LEXIS type systems was initially ad hoc using a variety of computing techniques - the full text approach, the concept indexed approach etc. It was only with Tapper's research into the elegance and usefulness of these different techniques that real progress was made. Bing and Harvold noted Colin Tapper's contribution to analyzing these approaches:

"The value of Tapper's experiments lies in two dimensions: firstly, the results he arrived at are of importance in themselves. Secondly - and in retrospect more important - the attitude Tapper brought to the field: the attitude of a critical and scientific

inquirer who examined the basic problems of designing reliable tests for information system performance."

[Bing, p67, 1977]

Using these quantitative methods, the field of legal information retrieval has progressed substantially. Of course, the problems are much simpler than those met in consultation systems - there is no normalization, analysis of concepts etc., only string processing and indexing techniques are used, (but see the proposal for "conceptual information retrieval" in [Schank, 1981] and also [Council of Europe, 1982]).

Mehl's second type of law machine has not been quite so successful or as easy to achieve as he imagined. It is, some might claim, useful to see the area in that same initial state as legal retrieval systems were before Tapper's quantification work - there are several approaches currently being taken, each working with different basic assumptions and techniques. Whether any approach currently being taken will be successful, though, is a matter for time and debate. None of the approaches are anyhow yet close to useful commercial implementation, apart, perhaps, from the "algorithmic" approach taken by the Inverclyde project [Addler, 1975] which is applicable to only a very small area of legislation (i.e. administrative law which, to a high degree, has been designed to be handled by rote, without recourse - normally that is - to the common law and which can be represented by simple data processing programming techniques). This thesis discusses the problem of representing general legislation and also the essentially computing problems which have been met by researchers in this area; also we describe the ELI program, which - when analyzed in terms of success and failure - is one source of information on possible routes for future expansion and research. As a brief introduction to some of the other work which has been done in the area though, we might think of McCarty's

TAXMAN project [McCarty, 1980a] a project which has been running for some considerable time using "semantic" or associative networks to represent legal objects and relationships; Stamper's LEGOL project has been mentioned above; JUDITH [Popp, 1975] which uses a hierarchical representation of law; and another researcher who uses associative networks is Meldman. Meldman's system attempted to match the user's input with models of assault and battery held in the network data structure.

We hold that some of the projects which we describe have approached the problem of legal knowledge engineering without a true appreciation of the jurisprudential and logical/legal problems involved. In this text we try to right this fault by laying an appreciation of some of the pitfalls which might affect the successful design of any legal consultative system.

## CHAPTER 3

### THE KNOWLEDGE BASED FRAMEWORK

#### 3.1 LEGAL KNOWLEDGE ENGINEERING

The development of computing techniques, described as Knowledge Based, which allow programs to encapsulate the "expertise" of a professional - say a doctor, or geologist - has given rise to the hope that such techniques may be capable of handling legal expertise in a manner which will allow the construction of high quality legal advisory systems. It is to this general goal that this thesis is directed and, in part, the thesis describes a program which gives advice on welfare rights legislation. Notwithstanding the various projects which are currently underway, in this early stage of the research field there are still many problems and many confusions. Broadly these problems can be divided into two categories. The first is the problem of legal knowledge engineering - that problem of ascertaining

- (i) just what it is in "law" that we wish to represent and
- (ii) how it can best be represented in terms of software.

Of these two knowledge engineering subdivisions, the first is philosophical (or rather, in legal knowledge engineering it is jurisprudential) and the second is the concern of computer science; these two subdivisions are the mainstay of this thesis. I argue for the need to see "law" as more than a formal logical system and also describe



the first tentative steps in building a useful knowledge engineered system.

The second major problem is that of man/machine interaction; this can be described as determining just how these systems will be used and by whom they will be used. These problems are briefly examined in Chapter 8; but although relegated to a later part of this thesis, it should not be assumed that this area is less important than the legal knowledge engineering aspects. In fact, our view (explained later) is that no successful legal or commercial implementation of these systems will be possible until the problem of the correct design for man/machine interface is properly solved.

The power of knowledge based programs is to be found in their ability to provide consultative advice to users. Simplistically, they encapsulate the "commonsense" but specialized expertise of "an expert". Thus, ideally, non-experts can access advice from the program at will. These knowledge based programs have, so far, been developed in areas where human expertise is scarce and expensive; it is in these areas that the prime motivation is perceived for the implementation of computer systems - the financial cost of consultation can be much reduced. Hopefully as well, if the cost of good professional advice is substantially reduced, then the professional users of such systems can increase the level of proficiency in their service to the client. These pressures of cost and quality of service have so far been mainly seen in the area of medical applications; for example, there has been a substantial amount of both statistical and technical input to the medical profession to achieve reductions in cost (more specifically, in terms other than the simply short-term) by more proficient health care. These same pressures exist upon the lawyer. In order to improve the efficiency of what has been called the "lawyering process" [Slayton, 1974] the lawyer, some have stated, has had to revise his or her research methods:

"The exponential growth in the volume of legal materials has dramatic consequences. First of all, it becomes extremely difficult to find all or even most of the materials relevant to a given problem, simply because so much may be relevant; accordingly lawyers have increasingly come to have an inadequate grasp of what they need to know properly to deal with the issue before them. The result is a 'deprofessionalization' of law."

[Slayton, p7, 1974]

As an example of this, few lawyers today can successfully handle any court case outside of their own speciality which requires searching of legal texts for potentially useful precedents (rather than those which are currently held to be useful) without recourse to a computerized retrieval system. Before the introduction of these legal information retrieval systems, the legal search process was described, as mentioned above by Kelso.

Since one of the weaknesses of these retrieval systems ([Slayton, p25, 1974] lists other reservations he has about them) is their inability to interpret the discovered texts, it would seem that the next logical step in the provision of computerized tools to the legal profession is the provision of knowledge based systems, which could provide some interpretation of legal texts and case precedents. We hold that the extraction of legal rules from the legislation - i.e. the interpretation of legal texts - is the primary legal task. As Twining and Miers state:

"Rule-handling is only one aspect of the crafts of law. Furthermore, interpretation is only one aspect of rule-handling. But it is basic - first, because most rule-handling activities involve or presuppose it, and, secondly, because a clear understanding of what is involved in interpretation inevitable throws light on a number of other matters as well."

The knowledge which we expect a legal knowledge based system to contain is the output from the expert's interpretation of legislation and precedent combined with his or her own "expert" view of the area.

A knowledge based system in the legal field could, at the most optimistic, be expected to aid the lawyer in understanding the relevant legal sources which the information retrieval system might point to. Alternatively, the knowledge based system could operate without contact with a retrieval system advising the lawyer on how to present a given defendant's, appellant's or prosecutor's case. In both cases, it would do so by some process of rule-handling; but, we emphasize and argue later - Chapter 9 - that rule-handling is not as simple a task in the legal world as it might at first seem.

Given the potential power and advantages of such systems (for instance, their use by advisors with limited legal expertise or in remote legal offices for handling slightly exotic problems), why are no such systems currently operating?

The reason for the non-existence of such systems is due to the complexity of the legal process; and it is to this problem that legal knowledge engineering must turn its attention.

Sections of this text examine "the law" to try and discover what the legal knowledge that we must handle actually is. As the ever continuing arguments in jurisprudence show, what this is is a moot point. What is accepted though is that the judicial decision is at the heart of the legal process; even in those cases which never come to court, the police (or - in Scotland - the procurator fiscal's) decision whether to prosecute, all assume that the judicial decision is the final arbiter of "the law" and that it overshadows all other aspects of the law. Levi pointed to the power of the judicial decision (albeit within the

framework of government):

"What a court says is dictum, but what a legislature says is a statute."

[Levi, p6, 1949]

and Hart spoke of the rules of adjudication which give the judge his power:

"... have conferred upon judges, where they have ascertained the fact of violation, the exclusive power to direct the application of penalties by other officials."

[Hart, p95, 1961]

We therefore examine the judicial decision and assess how we might be able to capture the essence of the legal process through the judicial process. The theoretical position which we shall take upon just what the judicial process is, is described later; for now it is sufficient to say that we lie at neither end of the continuum which says "law is logical", nor "law is classificatory". We rather take the Austinian view that problems of "law" are problems of power relationships evidenced in the language of judicial and social interaction: legal rules are the "language" of law, not its substance.

We accept that we (and other researchers) are some distance from the final and commercially implementable results of this legal knowledge engineering process, but we do believe that it is an achievable goal towards which to work; the hope is of advice easily accessible to even the smallest legal office allowing even the esoteric case to be competently handled in the most efficient manner with the maximum client satisfaction. Of course, good advice may simply be to tell the client that he has no hope of winning his case - good advice does not necessarily mean advice which always wins.

### 3.2 INTELLIGENT KNOWLEDGE BASED SYSTEMS

During the last decade or so there has been a dramatic shift in emphasis in research being undertaken by the Artificial Intelligence (AI) community. This shift is best seen as a move from abstract research dealing with the relationship between human cognitive processes and machine intelligence which we might describe as "machine learning" or "machine intelligence"; that latter described by Minsky as:

"The science of making machines do things that would require intelligence if done by men"

[Minsky, 1968].

The early AI research problems, usually based on game playing, were attractive for research: the rules of games could be clearly defined, success be measured (e.g. the chess program to beat a grand master) against a standard, heuristic solutions to combinatorial explosions can be compared. They were seen as problem areas which simulated problems in the real world. By solving these in "human like" ways, it would be possible to discover how these human problem solving processes operated.

The ultimate goal, certainly of the early research workers, was to be the General Problem Solver (GPS) - a device which by following defined "thinking procedures" could be applied to problems of all sorts - the general problem. The GPS programs developed (for example [Newell, 1972]) were based upon goal directed plans and operations; however, as Raphael notes:

"The fact that the GPS program successfully solved problems from several difficult task domains is at least as much a tribute to the cleverness of the programmers who encoded those tasks into GPS's object, operation and difference terminology, as it is a tribute to GPS's true generality."

[Raphael, p145, 1976]

The translation of these programming techniques to descriptions of how human cognition operates, though, has been adjudged unsuccessful, both by critics of AI:

"On my view the organization of world knowledge provides the largest stumbling block to AI precisely because the programmer is forced to treat the world as an object and our know-how as knowledge.

Looking back over the past ten years of AI research we might say that the basic point which has emerged is that since intelligence must be situated it cannot be separated from the rest of human life."

[Dreyfus, p203, 1981] (author's own emphasis)

and also members of the AI community:

"... general problem solvers are too weak to be used as the basis for building high performance systems. The behaviour of the best general problem-solvers, is observed to be weak and shallow, except in areas in which the human problem-solver is a specialist."

[Feigenbaum, p187, 1971]

The new research effort is directed into how techniques gleaned from earlier research (for example, work into heuristic control of search strategies) can be utilized in practical systems. One example of this shift is in the application of "expert system" techniques [Michie, 1979] to real world situations, rather than the emphasis on "micro-world" problems, whose toy-like aspect has been described by their proponents:

"Each model - or "micro-world" as we shall call it - is very schematic; it talks about a fairyland in which things are so

simplified that almost every statement about them would be literally false if asserted about the real world."

[Minsky and Papert, p39, 1970]

These expert systems have, after the Alvey committee [Alvey, 1982] become known as Intelligent Knowledge Based Systems (IKBS). We shall use this term here, except where use of "expert system" is more illustrative.

There are, it is claimed, a growing number of these IKBS programs in operation at present (leaving until later just what is meant by 'in operation'). For examples see DENDRAL [Buchanan, 1969]), PROSPECTOR [Duda, 1979].

Research into IKBS is currently at a high pitch, and much is being expected of it - for example one book review has recently stated "A New York market research firm has recently announced that the artificial intelligence market in the United States will rise slightly over \$155 million in sales to \$2 billion by 1989". We can see this research interest from looking at one of the most discussed IKBS programs - MYCIN [Shortliffe, 1975]. Interest in this system is due in part to the fact that it has been extended to allow teaching with NEOMYCIN [Clancey, 1981], easy debugging of the knowledge base with the TIERESAS system [Davis, 1977a]; it has also been stripped to provide an expert system "shell" in the EMYCIN system [Van Melle, 1981]. A "shell" consists of a rule interpreter and any other data/processing structures which might be used to interpret a rule corpus. The rule corpus is supplied by the user of the shell. The shell is thus a means of allowing one program to give consultative advice in more than one domain. It has been found, though, that such a shell is not always sufficient; Rooney has noted that many researchers consider currently available shells to be the:

"lowest common denominator of the expert systems work over the last

10 years."

[Rooney, 1984]

TIERESAS has also been used to test knowledge structuring techniques (i.e. "meta-knowledge" - that is knowledge that the program possesses about its own problem solving knowledge [Davis, 1980a] [Davis, 1980b]).

Although most IKBS programs utilize a variety of data structures (for example, MYCIN uses rules and a tree structure - the "context tree" - as its prime data structures), there has been a tendency to use a production system methodology as the basic data driven processor. There are a variety of early applications of this formalism; Post [Post, 1943] was perhaps one of the earliest, but, due to their usefulness, production systems have appeared in other applications - Chomsky's 'Syntactic Structures' [Chomsky, 1957] illustrated how a grammatical theory could be based upon them (most computer languages now have their syntax specified by means of a production system - for example see the Pascal Report and User Manual [Jensen, 1975]).

As an indication of how the AI community has seen these new IKBS techniques as a radical alteration in their technical framework, we might note Newell and Simon's heady description of the production system formalism as:

"One of those happy events, though in minor key, that historians of science often talk about: a rather well-prepared formulation sitting in wait for a scientific mission."

[Newell, 1972]

We do not closely examine the production system formalism (but see, for an introduction to the "traditional" production system architecture [Davis, 1976]), but in Chapter 4 we see how it might be used to model a judicial justification.



### 3.3 PERCEIVED ADVANTAGES OF IKBS TECHNIQUES

It is useful to uncritically view IKBS techniques, to see why their perceived advantages have been considered useful to so many applications, including the commercial and industrial:

"The development of such knowledge based systems is widely regarded as the best means of expanding the application of IT to activities which today's computing technologies cannot approach."

[Alvey, 1982]

Our position relating to these claims will become clear throughout this text; for the moment it is out to be stated that we are sceptical about many of the claims made for IKBS techniques and completely sceptical about all the claims made for artificial intelligence.

An IKBS system can be thought of as a black box which accepts knowledge from one or more expert and "intelligently" offers advice to a non-expert. Thus the alternative descriptor, expert system, is explained by the fact that the system contains an expert's expertise.

There are certain areas where these IKBS systems are effective - primarily areas where problems to be solved rely heavily on abstract, "commonsense" knowledge, rather than well defined "algorithmic" solutions. For example, note the following interaction which one researcher has proposed as a goal to arrive at between expert system and man:

Expert: The pump pulley should be next.

Novice: Yes--uh, does the side of the pump pulley with the holes face away from the pump or towards it?

Expert: Away from the pump.

Novice: All right.

Expert: Did you insert the key -- that is, the half-moon shaped piece?

Novice: Yes, I did.

Expert: Be sure you check the alignment of the two pulleys before you tighten the set screws.

[Hart, p832, 1975]

Of this dialogue, the author writes:

"This fragment illustrates several important abilities of consultants that contrast sharply with a static information source like a manual. First, notice that a question from the novice is answered directly and in his terms. There is no need for him to search through a mass of information, or to convert information from an abstract or "standard" form into a directly usable form. Notice that the expert is checking on progress by offering warnings and reminders about critical steps. This has the function of not only minimizing errors, but also of allowing the expert to keep track of the progress of the work. The latter function is the basis of the expert's ability to present relevant advice, and to put it in a context that is familiar to the novice."

[Hart, 1975]

IKBS systems claim their effectiveness from the fact that they can embody the expert's knowledge produced by interaction between the system and expert (ie they are "taught"). This knowledge once embodied becomes explicit as opposed to implicit - ie hidden - within an algorithmic program's coding, and can be used to "explain" its reasoning. For example:

"A consultative production system need not be a psychological model, imitating a human's reasoning process. The important point

is that the system and a human expert use the same (or similar) knowledge about the domain to arrive at the same answer to a given problem. ... The process of trying rules and taking actions can be thought of as "reasoning", and explanations consist of showing how rules used information provided by the user to make various intermediate deductions and finally to arrive at the answer."

[Scott, 1977]

and other writers have also shown the importance of self-explanation:

"Computers can be inscrutable. To the layman, the computer is often regarded as either an omniscient, fathomless device or a convenient scapegoat. In part, this situation has arisen because computer systems are designed with little provision for self-description. That is, programs cannot explain or justify what they do. ... Trust in a system is developed not only by the quality of its results, but also by clear description of how they were derived."

[Swartout, 1983]

The explanatory abilities of these IKBS programs has also led Duda et al to suggest Prospector's role in tutoring:

"In addition, most of the geologists we know who have had any experience with Prospector have remarked about its potential value as an educational tool. In this regard, the models in the system contain explicit, detailed information synthesized from the literature and the experience of expert explorationists, together with explanatory text that can be obtained upon request. Furthermore, a typical consultation session with Prospector costs only about \$10 at current commercial computer rates."

[Duda, 1979]

The expert builds up a Knowledge Base (KB) within the system; as the KB is being developed the expert can test the validity of the input information. In effect he asks the system questions (as a teacher asks a pupil) to test its understanding of the given problem (although - we believe - this is more akin to the debugging process in 'traditional' programming). Eventually (or hopefully so) the system will have 'learnt' from the expert enough knowledge to allow it to intelligently react to problems which it is set in its knowledge sphere. At this point it is possible to use another expert (from the same field) to interact with the system and further verify and expand, if possible, the system's understanding, allowing a higher level of machine competence than that provided by the individual experts. This knowledge, held in the KB, can then be the basis for consultation between the system and the non-expert. Thus the expert's advice is readily available in his/her absence.

Such then is the perceived position of expert systems: it is, we believe, a naive view of what can be achieved from contact with a computer. Hopefully the results of analyzing the ELI program will bring some of this hyperbole to ground.

### 3.4 ARTIFICIAL INTELLIGENCE, LEARNING AND KNOWLEDGE

So far, terms like "artificial intelligence", "learning" and "knowledge" have been used without proper definition of just what is meant. Such value loaded terms might cause an undesired interpretation to take place. How, in this text, should we approach such use?

The problem arises because the IKBS research field has, at least until now, been developed by workers from the AI and cognitive psychology fields (for example, Stanford's Heuristic Programming Project and MIT's Artificial Intelligence Laboratory) and has taken the terminology of psychology rather than that of computer science.

From the framework of this research, perhaps the best (critical) introduction to the fields of AI, machine learning and psychological modelling is the Lighthill Report [Lighthill, 1973], incorporating as it does, Sir James Lighthill's views and retorts from AI researchers (for example Prof. Donald Michie). The perspective we have taken in this research is akin to Lighthill's, that is scepticism about the "learning" or "intelligent" abilities of computer programs (see, for example, the effective critique of AM's learning ability by [Ritchie, 1984]).

Lighthill wrote:

"To sum up, this evidence and all the rest studied by the present author on AI work within category B during the past twenty-five years is to some extent encouraging about programs written to perform in highly specialized problem domains, when the programming takes very full account of the results of human experience and human intelligence within the relevant domain, but is wholly discouraging about general-purpose programs seeking to mimic the problem-solving aspects of human CNS [central nervous system] over a rather wide field. Such a general-purpose program, the coveted long-term goal of AI activity, seems to be as remote as ever."

[Lighthill, p15, 1973]

We would argue that this statement still holds true.

Taking this view and extending it into the IKBS field, we would say that IKBS systems do not exhibit "intelligent behaviour" - rather that they are clerical systems which operate upon data which they do not "understand". We might call the data "knowledge" but the case that machine representations of knowledge are human-like is unproven, although many use descriptive language which suggests the machine 'reasons'; for example:

"From its beginnings, artificial intelligence has borrowed freely

from the vocabulary of psychology. The use of the word "intelligence" to label our area of research a case in point. Other terms referring originally to human mental processes that have currency in AI are "thinking," "comprehending," and, with increasing frequency in the past five years, "understanding." In fact, these terms are probably used more freely in AI than in experimental psychology, where a deep suspicion of "mentalistic" terminology still lingers as a heritage of behaviourism."

[Simon, 1977]

The links between the disciplines of AI and cognitive psychology are and have been strong (for example, much work in Artificial Intelligence done at the Open University is carried out under the auspices of the psychology discipline rather than within the computing discipline). Yet another example of the strength of the links between the two disciplines has been the historically important question: "How closely related is machine intelligence to the human thinking mechanism?" This relationship has been set for almost all early AI programs, for example:

"Simulation of human intelligence is not a primary goal of this work. Yet for the most part I have designed programs that see the world in terms conforming to human usage and taste."

[Winston, 1975]

But the question is complex - if a chess program consistently beats a grand master (or, in fact, novice) does this mean that the machine "thinks" about the game in a better way than the grand master? It seems to us that no commonality between machine and thought has yet been proven.

Ades has noted another problem:

"Everyone knows that for people who are trying to build computer models, to refrain from adding a few kludges to the program to make it 'work' is more than most mortals can do. Even more obvious is the distinction between an effort to understand a natural phenomenon and a program that mimics without trying to model the natural processes themselves. What is less to be expected is that psychologists should take these programs seriously as insights into the workings of the human mind."

[Ades, 1981]

As well as analytical comparisons between psychological theories and coded programs, programs have also been developed as a direct implementation of aspects of psychological theory. For example, in the majority of AI programs (including IKBS programs) a major element of the design has originated in intuitive or introspective models of the human thinking mechanism. Minsky's frames are used by many researchers; note how Minsky inter-relates the human "mental" action with the computing idea of "data structure" when outlining his frame theory:

"Here is the essence of the theory: When one encounters a new situation (or makes a substantial change in one's view of the present problem) one selects from memory a substantial structure called a frame. This is a remembered framework to be adapted to fit reality by changing details as necessary.

A frame is a data-structure for representing a stereotyped situation, like being in a certain kind of living room or going to a child's birthday party."

[Minsky, p212, 1975]

The ELI program described in this text is based upon the production system formalism used by Newell and Simon [Newell, 1972] as a method of human problem solving. By examining the way in which subjects solved certain problems Newell and Simon were able to provide production models which explained their explicit actions. O'Shea and Young [O'Shea, 1978] have similarly researched errors made by children when learning arithmetic.

Newell and Simon saw a close inter-relationship between the production formalism and human cognitive processes:

"We confess to a strong premonition that the actual organization of human programs [sic] closely resembles the production system organization. .. We cannot yet prove the correctness of this judgement, and we suspect that the ultimate verification may depend on this organization's proving relatively satisfactory in many different small ways, no one of them decisive."

[Newell, p803/4, 1972]

We are sceptical about, at any rate, the traditional production system being a satisfactory or sufficient model of behaviour. It is a suitable model of problem reduction (as noted for Prolog - an "extended" version of the production system architecture - by [Kowalski, 1979]) but cannot handle a variety of other kinds of problems, for instance the use of negative knowledge (that knowledge where we say, such-and-such is not a form of the other such-and-such). Also, as noted by Goldstein and Grimson in their work on flight control, the production system formalism is incapable of encapsulating all desired information:

"The assumption that all rules are independent carries with it the additional assumption that all rules are equally likely to be used at any stage of operation. In this case, since all rules are sensitive to context, such an assumption is not valid. ...



"Another common problem associated with production systems is the 'implicit context problem'. This is the fact that the rule base has a total ordering associated with it and the position of the rule in this ordering becomes an important factor. Thus, since a rule ordinarily won't be called unless the rules preceeding it in the total order have failed, there are in essence extra conditions on the application of the rule. This may affect the performance of the system."

[Goldstein, p313, 1977]

To handle the information we might wish, other more complex - or "bastard" - production systems are essential. The ELI program therefore tried to extend the traditional production system architecture by, for example, incorporating annotations.

Some scepticism about "mental modes" and their relationship to computer science is needed - our intuitive insights into these models and theories are often wanting and misleading. Psychology itself being less than clear on its subject matter, we would do well to agree with Wittgenstein:

"The confusion and barrenness of psychology is not to be explained by calling it a "young science"; its state is not comparable with that of physics, for instance, in its beginnings. (Rather with that of certain branches of mathematics. Set theory.) For in psychology there are experimental methods and conceptual confusion. As in the other case conceptual confusion and methods of proof.)

The existence of the experimental method makes us think we have the means of solving the problems which trouble us; though problem and method pass one another by."

[Wittgenstein, p232e, 1958]

Therefore, this text does not try to relate "program" to "mind" - it is not a work on cognitive (or other) psychologies.

## CHAPTER 4

### LOGIC, FORMAL MODELS AND LEGAL REASONING

#### 4.1 INTRODUCTION

If logic has no place within the law then it must be assumed that any attempt to produce a computerized legal consultative system, that is a system which can present legal advice rather than textual information, must be doomed to fail. We mean, of course, by "legal advice" the sort requiring the intelligence which researchers in AI are working towards: clerical advice (advice by rote) is much less than that since it does not have the ability to deal with "new" legal situations and we hold that most non-trivial legal enquiries have some "new" aspect to them. Such an advisory system was proposed, perhaps rather optimistically, by Mehl with the statement:

"One may also conceive - more ambitiously - a consultation machine which will answer any questions put to it over a vast field of law."

[Mehl, p768, 1958]

Mehl's proposition is untenable if logic has no place within the law because a computer is basically a machine for explicating a logical system. Thus any formalized logical system can be computerized, and only such logical systems can be computerized. However, if logic does have a place within the law (by which we mean that some deductive

process operates in the judgement of a particular case) then Mehl's conjecture may well be realizable.

In this chapter, we attempt to show that there does exist a logical system which illustrates what might be the deductive process whereby a judge uses logic to arrive at a judicial pronouncement; that is, that legal reasoning may be deductive, rather than inductive as proposed by several researchers. To carry out this logical modelling it is necessary to:

- (i) have a correct understanding of the problems of discussing "logic",
- (ii) have an understanding of how 'logic' can relate to 'law',
- (iii) present a logical model which mirrors - in part - legal reasoning,
- (iv) discuss how close an affinity might exist between the presented logical system and a judge's actual decision process.

We shall propose, in fact, that our deductive model falls some distance from our desired goal. In this chapter that failure will be shown by recourse to logical problems; later, it will be shown by Realist arguments.

#### 4.2 THE NATURE AND ROLE OF LOGIC.

It is regretful that, in a discipline which deals with the philosophy of law, when discussing the role of logic within the law researchers have tended to assume that logic is a relatively simple (if diverse) subject. But as Guest states:

"What do we mean by logic? Too often discussion of this subject has centred around the rather barren controversy whether legal reasoning is deductive or inductive in form. In this dispute, both sides have assumed that by deductive logic is meant the most simple form of Aristotelian syllogism: All S is P; x is an instance of S;

therefore x is an instance of P."

[Guest, p181, 1961]

In the following sections, we try to illustrate the more subtle aspects of logic. For instance, one of the principle confusions in discussions about logic's role with regard to the law, is just what relationship "logic" has to notions of 'correct argument'. Before dealing with formal logic, though - that is both the syllogism and its extensions and second order systems - it is essential to clarify what MacCormick [1978, p38] has called the 'everyday' form or usage.

#### 4.3 EVERYDAY LOGIC

A term which is used both colloquially and with such deep philosophical dimensions as "logic" is bound to confuse its users, especially if both conceptions are being employed at the same time. In the following sections we attempt to clarify the technical perspective with which logic is applied, divorcing it from extra-systematic conceptions of truth and validity, and ignoring the colloquial aspects. By extra-systematic is meant the relationship which a logical system has to what is, technically, outside its formalism; thus "thought" stands in an extra-systematic relationship to the syllogism. In many ways the real problem in discussing logic and the law is caused by such colloquial and technical confusions. MacCormick noted a distinction between formal logic (or in his case *modus ponens*) and everyday logic:

"But is it really logical that a decision should be given against Mrs. Tarbard? somebody might object: 'It's not logical at all to hold the publican liable, when she is in no way responsible for the contents of a sealed bottle of lemonade, and especially not when the manufacturer is, as in the present case, absolved from liability - since he after all is the only person who has any control over what gets into the lemonade bottle before it is

sealed.' It will be observed that such an objection echoes the regret expressed by Lewis. J in granting judgement against Mrs. Tarbard who was, as he said 'entirely innocent and blameless in the matter'."

[MacCormick, p37, 1978]

Thus the question which MacCormick raises is: how can logic be non-logical. How can (what MacCormick believes to be) a logically justified legal decision be illogical? He answers the question by stating that there are at least two different senses to the word logic. The first of these is where,

"an argument is logical if it complies with the requirement of logic, that is to say, if its conclusion follows necessarily from the premisses"

[MacCormick, p38, 1978].

And the second is the "everyday usage":

"some action or state of affairs can be said to be "illogical" in that it "doesn't make sense"

These two different senses are

"only partially overlapping."

[1978,p38]

The essence of "everyday logic", to MacCormick is that within its scope an argument, if it is called illogical, has the quality that it does not make sense. Another view is that everyday logic can be seen in a much stronger light; it is prescriptive in that rather than simply not making sense, such a description as illogical implies an "ought" or a "should"; that the logical resultant should or ought to be something else. Thus that it is illogical that a decision should be given against Mrs.

Tarbard, carries the implicit assumption that another result (i.e. that a decision should not be given against Mrs. Tarbard) ought or should be found.

In this view then, "everyday logic" involves patterns of belief - both individual and social (because it is possible that only Mrs. Tarbard could have thought the decision against her illogical, or that 'everybody' thought it illogical). When our expectation of the operation of the real world (or the legal world) is contrasted with actions which occur, we necessarily have a view of what the results "ought to be".

The wider consequences of such a perspective of everyday logic are that, in effect, it has little to do with formal logic at all. Rather it is a social process which has a tendency to state expectations of normative behaviour.

#### 4.4 LOGIC AND TRUTH

The basis of Aristotle's logic was the assumption that valid argument could be based upon one elemental form - that of the proposition where a subject and a predicate are joined by a connective. An example of such a proposition is, "a dog is chewing". To Aristotle, this proposition's usefulness is that it can be proven or disproven - either the dog is chewing or it is not; the proposition is either true or false - due to the law of the excluded middle. The law of the excluded middle simply states that no "middle" possibility can be encountered; for example, there cannot be True, False, and Possibly True.

The format of valid Aristotelian propositions can be seen as (where s is the subject and p the predicate):

- (i) all s is p
- (ii) no s is p

(iii) some s is p

(iv) some s is not p

By combining two of these propositional forms, a valid conclusion can be reached:

(a) all s is p

(b) all x is s

(c) all x is p

But not the following:

(a) some s is p

(b) some x is s

(c) some x is p

The discovery of just which combinations of propositions can provide a valid syllogism is one of Aristotle's achievements. That it was an achievement can be realised by the fact that for two thousand years Aristotle's codification of valid forms of deduction was regarded as complete and incapable of essential improvement. As late as 1787, Kant was able to say that since Aristotle formal logic

"has not been able to advance a single step, and is to all appearances a closed and complete body of doctrine".

Kant, was wrong as we now know; Nagel and Newman state:

"The fact is that traditional logic is seriously incomplete, and even fails to give an account of many principles of inference employed in quite elementary mathematical reasoning. For example, of the principles involved in the inference: 5 is greater than 3; therefore, the square of 5 is greater than the square of 3".

[Nagel, p40, 1971]



Aristotle's syllogism was not a dry and abstract formal theory; it is important to realize that Aristotle was attempting to describe "correct argument" in much the same way that the Platonic School believed that their geometrical discoveries were discoveries of absolute truth. Kline states:

"Plato said, further, that by using complicated instruments 'the good of geometry is set aside and destroyed, for we again reduce it to the world of sense, instead of elevating and imbuing it with eternal and incorporeal images of thought, even as it is employed by God, for which reason He always is God.'

[Kline, p70, 1972]

Such an attempt to discover "true reasoning processes" involves distilling the essential nature of discourse into a formalism so that its truth or falsity may be ascertained. Such a view of logic is still consistently found today. One example is Ilmar Tammelo:

"Explicit and sound knowledge of logic brings many benefits to the lawyer. Above all it helps the reasoner to acquire proficiency and self-confidence in reasoning. For logic charts the practicable roads of reasoning and the pitfalls which await those who diverge from these roads. Those who master the principles and methods of logic are capable of quickly discovering valid arguments, defects in the reasoning of their own as well as their opponents, to disclose flaws in any discourse and to dispose of or overcome them efficiently. This gives poise to the reasoner in all argumentative situations."

[Tammelo, pIV, 1969]

Another is MacCormick who states:

"It should not be forgotten that the main and perhaps the only point of technical symbolisms and notations is to exhibit and make plain possible forms of argument and conditions of valid argument..."

[MacCormick, p286, 1982]

And Boole spoke of the "science of the intellectual powers":

"Let the assumption be granted, that a science of the intellectual powers is possible ... "

[Boole, 1854]

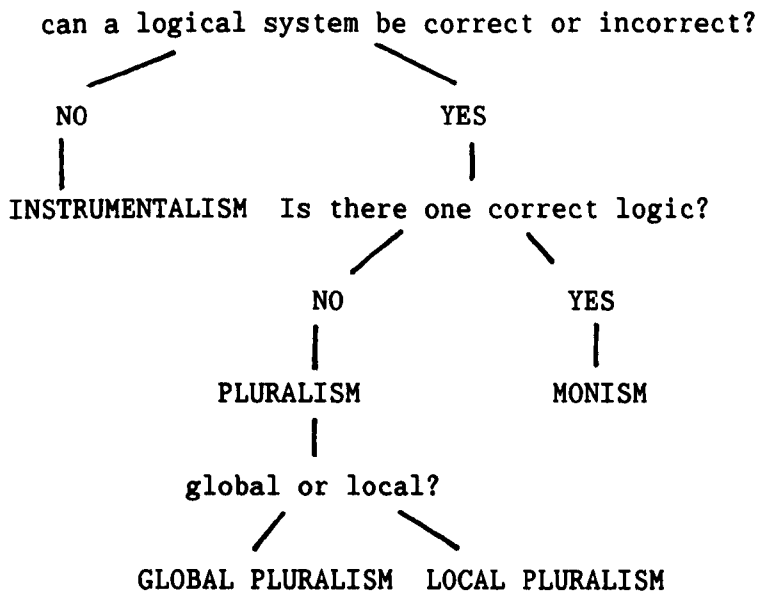
where logic was to be the form of these intellectual powers.

Today, Aristotle's conception of logic is thus contentious, necessarily subsuming within its definition notions of logical systems as "correct" or "incorrect" with respect to notions of absolute truth. Such notions are to do with the relationship of a logical system to the extra-logical world. Haack summarizes the issues as:

"Does it make sense to speak of a logical system as correct or incorrect? Are there extra-systematic conceptions of validity/logical truth by means of which to characterize what it is for a logic to be correct? .... Must a logical system aspire to global application, i.e. to represent reasoning irrespective of subject matter, or may a logic be locally correct, i.e. correct within a limited area of discourse?"

[Haack, p225, 1978]

Haack clarifies the theoretical positions with the following diagram:



Briefly, the instrumentalist believes that it is unimportant if a logical system has extra-systematic validity; rather that a logical system is a guide which refines predictive abilities, just as the laws of physics are not "correct" - only principles of inference. (Although, of course, there are various positions of instrumentalism - one view allows intra-systematic truth ("logical-truth-in-L"); another that logic is not a set of statements, rather a corpus of procedures). From this view, logic is a "model" of the discourse it deals with - it may be a 'good' model, in which case it reflects well that which it models, or it may be a 'bad' model in which case it reflects poorly that which it models. The opposing view (taken by both the monist and pluralist) is that logic can be much more than a model; that a logical system has extra-systematic validity and is somehow an intrinsic element of the real world. Thus the modus ponens:

s then p  
 s  
 ∴ p

contains the essence of a good (i.e. valid) argument, unlike

s then p  
q  
∴ s

which contains the essence of a bad (i.e. invalid) argument. Since instrumentalists must be expected to see the latter as bad or an example of false reasoning, they argue against extra-systematic validity by reference to other aspects of the syllogism and its extensions. A primary fault is found in the scope of the variables i.e. what can be incorporated within the class s or the class p. The idea of the syllogism dealing with classes arose with Boole's work in the 19th century. As mentioned later, there are arguments against such a view. The problem has been stated in one criticism of the application of modus ponens to legal argument as:

"with some purportedly syllogistic arguments about 'objects' we are confronted with the question whether genuine propositions are conveyed in the words used"

[Wilson,1982,p274].

Such an argument can be seen by recourse to the following example. Assuming for the moment, that the modus ponens form:

if s then p  
s  
∴ p

is a valid reasoning process, i.e. that it explicates extra-systematic reasoning, we might try to argue that if the variable s is given the value (or make it refer to the proposition) "dogs bark" and p the value (or make it refer to the proposition) "cows moo". When the form is restated, it becomes a nonsensical:

If dogs bark then cows moo

dogs bark

∴ cows moo

This is not nonsensical because of the inherent logical form - only because we know that there is no implicative link (as suggested by the proposition) between dogs barking and cows mooing in 'real life'. However, the propositions in the above argument are easily related to the real world - everyone knows that dogs bark and cows moo - but replace s by "sold by description" and p by "goods should be of merchantable quality" and the inherent problems of such logical form are focussed:

If sold by description then goods shall be of merchantable  
quality

Sold by description

∴ goods shall be of merchantable quality

The modus ponens does not clarify the concept "sold by description" nor "merchantable quality"; neither does it prove any implicative link between the two - so how can we be sure that the argument has "any real world validity". Such faults and weaknesses of the syllogistic form and its extensions caused it to fall into disrepute in the medieval period. Kilminster described this medieval scepticism over the syllogism and also one reason for its survival:

"The whole matter is trivial and the reader may wonder ... why it has survived for so long ... we find in the statutes of the University of Oxford in the 14th century the rule "Bachelors and Masters of Arts who do not follow Aristotle's philosophy are subject to a fine of 5s. for each point of divergence, as well as for infractions of the rules of the Organon". More generally, academic authority has had a great deal to do with the survival of this archaic structure."

#### 4.5 CONSISTENCY OF LOGICS

The syllogism and associated forms are but one type of formal logical system. They do, however, along with all first order systems (e.g. predicate calculus) have internal validity - they are consistent. Consistency is a mathematical concept which is related to the existence of paradoxes. Nagel and Newman [Nagel, 1971] present one of the best non-mathematical introductions to the idea of consistency and its importance for applications where extra-systematic validity is essential - e.g. logical positivism's need for a logically secure basis, as in [Wittgenstein, 1981]. A claim for a logic to have extra-systematic validity necessitates that that logic is free from paradoxes - e.g. that it cannot produce the formulae "p or not p" and "not(p or not p)" at the same time; if it can, then the system is inconsistent. Since the rationale behind such introduction of logic as a "correct form of thought" utilized the assumption that logic was a consistent body of knowledge, the proof that a logical system is consistent supports the 'logic as correct thought' proposition. This consistency of first order logics was established by Kurt Godel [Godel, 1930] and while not proving the validity of the logic as thought thesis, consistency does add weight to such a psychological theory. Haack's "monists" would stop at this point in their discourse of logic, there being they claim one correct logic. However "pluralists" (and most instrumentalists) would include other formal logical systems within their definition of logic. Such other systems are modal logic (which tries to escape from classical logic's two truth values - true and false - and include notions of true, might be true, etc.) and production systems (which deal with strings of variables) dealt with later in this paper. These systems, though, are not provable as consistent. Godel proved [Godel, 1931] that such systems require meta-systems to prove their validity; unfortunately such

meta-systems also require meta-systems to prove their validity and these systems, etc. ... It was in this paper that Godel undermined the philosophical basis of Whitehead and Russell's Principia Mathematica [Whitehead, 1910-13].

Such problems concerning the consistency of such second order systems tend to support the instrumentalist point that logical systems should be seen as models of the world rather than intrinsic aspects of the world.

#### 4.6 THE NOTION OF CLASSES

Another complex area for those wishing to apply logic to the law is the notion of a "class". Thus the syllogism:

All men are Mortal  
Socrates is a Man  
∴ Socrates is Mortal

can be seen in terms of classes (which is why logic is sometimes called the Calculus of Classes) or sets (as in Venn diagrams) of objects - thus 'Socrates is a man' does not apply simply to the individual "Socrates" but rather to that class or set of people known as "Socrates".

The proposition that modus ponens could be seen in such terms was first made in "The Laws of Thought" [Boole, 1854]. This view caused Boole to become known as the founder of modern logic, since such a perspective of logical systems is the predominating one at present. The notion of a "class" has also been utilized by those who claim that legal reasoning is not logical at all. For instance Jensen states:

"thus if a caretaker of a block of flats refuses to allow the residents to keep dogs on the grounds that they are a nuisance, the other residents would probably think it illogical if he refused to regard the dog of a blind resident as a nuisance. Suppose,

however, that the caretaker's reasons were that the dog enabled the blind man to find his way in and out of the building without his having to trouble others to help him. Would the caretaker be illogical? It is not logic - nor even abuse of logic - to insist on placing something in a certain class because of its resemblance in some respects to other members of the class when the basis of classification has not been explicitly formulated, and when so to place it would defeat the aims or the attainment of which the classification was made. There is no rule of logic against substantially contracting, or stretching a term."

[Jensen,p9,1957]

Is this action by the caretaker illogical in the formal sense? It does seem so at first impressions. However, when different assumptions are taken (e.g. that guide dogs provide a necessary service to the blind) it then seems reasonable (logical, in the colloquial sense) that such dogs are not forbidden from the premises. Might one not argue that this "classification" is the result of different assumptions; that the assumptions can be seen as propositions and that:

If a dog then precluded from flats

a dog

∴ precluded from flats

is very different from:

If dog is not a guide dog then precluded from flats

not a guide dog



∴ precluded from flats

While this does show that Jensen's example can be reformulated in terms of modus ponens, (although, note that MacCormick [1982,p289] suggests that the delineation of the class can be built up by a hierarchy of properly quantified steps. We tend to support this, if not the utility of quantification; and mention it later when proposing that Russell's conception of induction as guesswork and deductive justification for these guesses might be correct) there is another point about Jensen's contention of class which overtly permeates thinking and writing. The essence of the point is that it is wrong to build imaginary "sets" or "classes" which are delimited before they are filled - (e.g. Cantor's work with infinite sets) - such an attempt is tautologous in that the sets are recursively defined; they are what they are because we define what they are filled with because of what they are. While mathematically useful, they are philosophically unsound and confusing.

While really outside the scope of this paper, those dealing with logic and the law should be aware of such criticisms; especially when such criticisms are often virulently made - Wittgenstein, for example, described set theory as a cancer pervading mathematics and thought.

One result of this concept of logic as class or set based is the need, as some see it, to formalize the set within either existential or universal quantifiers - to create a world of possible members which can be incorporated within an encompassing universal set; for example:

" "For all people, if one person transfers the property etc., then a contract of sale exists" or symbolically,  $(x)(\text{If } Fx \text{ then } Gx)$ ."

[White, p742, 1980]

Such requirements when demanded of all logical systems strike us as being unnecessarily pedantic, if not erroneous; below, we detail a logical system which advantageously omits such quantification.

#### 4.7 APPLICATION OF LOGIC TO THE LAW

##### 4.7.1 Previous Applications

Given the very confusing and diverse views about logic, from both logicians and laymen, there might well be a tendency to hesitate conjoining logic with jurisprudence. Blatantly though, this is not so - all that has happened is that each writer on the subject has brought his own perspectives about logic to the application. For example, Del Vecchio hoped for a scientific formalism of legal theory using logic:

"Every educated man, it would seem, should be able to answer the question, "What is law?" with facile certainty, and his thoughts and acts should be controlled by its concept. But this is far from true; many of those who have given special study to legal principles and institutions would hesitate to offer an answer. Whoever considers the diversity and incongruity of its definitions cannot fail to see that, notwithstanding the labor expended on the science and philosophy of law and the number of special researches recently instituted, the concept of law has not yet assumed definite shape in a truly logical sense."

[Del Vecchio, 1914, p1-2]

Such a concept of law assuming an inviolate base is similar to the logical positivist view of logic as a methodology for understanding the underlying meaning of language; that is, an ideal language. Both are attempts to apply 'logic and the scientific method' to the social sciences. Such an approach was also taken by Kelsen in the proposal for a "pure theory of law". He stated:

"It is called a "pure" theory of law, because it only describes the law and attempts to eliminate from the object of this description everything that is not strictly law: Its aim is to free the science of law from alien elements. This is the methodological basis of the theory."

[Kelsen, p1, 1967]

Stone has mentioned that because Kelsen sometimes uses the concepts of logic in a loose manner does not mean that he did not try to aim for a logically consistent theory:

"It has to be said, indeed, that though many of Kelsen's basic positions are dependant on the cogency of inferences, his use of even basic terms of formal logic such as "contradiction" and definition is often loose. This, no doubt, explains why Kelsen's work is simply omitted from a recent competent and extensive bibliography of juristic logic."

[Stone, p135, 1964]

Other assumptions of what logic is, have centred around its usefulness as a tool to formalize jurisprudential concepts, clarifying the problems caused by "chameleon-hued words" as Hohfeld described them. For example, Layman E. Allen used formal logic to refine Hohfelds' "Rights" theory:

"With the advent of the digital computer and the power of electronic information retrieval systems, the precise usage and definition of words rises from the level of merely aiding the efficiency of a transaction between legal entities to that of being virtually essential, where computers are involved, if the transaction is to take place at all. Man learns by example and possesses the creativity to resolve ambiguities; ... In general, a

computer requires a clearer and more precise statement of the question to be resolved. The purpose of this Article is to examine one phase of the legal communication process - that of defining a legal relationship between parties - with a view toward expressing that definition in a manner that will facilitate careful and precise communication wherever that is deemed desirable".

[Allen, p429, 1974]

Others have combined the two perspectives (that of analysis and of theorizing); for example Reed C. Lawlor wrote:

"On the one hand the author is testing methods for analyzing and predicting judicial decisions. On the other hand he is testing a computerized theory of stare decisis including a theory of personal stare decisis."

[Lawlor, p219, 1980]

Such a dualistic approach is not uncommon, Abraham Fraunce proposed that logic underlay proper application of the law, and he also used logic to analyze a case concerning the Earl of Northumberland.

Our own work does not attempt to prove that logic underpins the law; neither does it attempt to produce a logical theory of the law. Our goals, in this chapter, are much more limited than either of these - namely to show that legal reasoning can, in some cases, be modelled by a logical system with the goal of translating this logical model into a computer system. That this attempt has usefully illustrated some contentious aspects of legal reasoning comes only from the general utility of such a model, especially when that model is implemented as a computer system and must stand on its own resources.

#### 4.7.2 A Logical Jurisprudential Model

There is a distinction between a "theory" and a "model" which makes us determine not to speak of a "Logical Jurisprudential Theory". The distinction is quite subtle but, we believe, important. A theory is an attempt to subsume all disparate facts about the object of study under one predictive explanation. A theory can be a precise explication (as we see Einstein's theory as) or can be coarsely explicated (as most jurisprudential theories are, utilizing arguable assumptions within their terms of reference - "right", "grundnorm", "coherence"). It must, though, by definition explain and predict in a consistent manner its whole area of discourse - however limited that area might be.

A model, on the other hand, is a tool - it need not necessarily explain or predict. Normally though, a model which neither explains nor predicts is of little analytical use. A model should, for our purposes, be useful; it should give insights into that which it models in much the way that a theory tries to give a complete insight.

A model usually attempts to model a subset of a theory; when it tries to model a whole theory the value of distinction between "model" and "theory" becomes hard to justify.

Thus: a theory of legal reasoning must aim to explain 'legal reasoning'; and a theory of law must aim to explain "law". But: a model of legal reasoning should give insight into legal reasoning - perhaps by clarifying one issue or proving that a proposition about legal reasoning is possible (it does this, of course, by being a successful model of that proposition); alternatively, the model might try to show strengths and weaknesses of the theory which are not obvious in the theory.

This approach to the usage of a model is described as a "simplifying model" [Enc. of Phil., Vol 5, p355, 1967]. In the next section we propose just such a model.

#### 4.8 A JURISPRUDENTIAL MODEL OF LEGAL REASONING

##### 4.8.1 Deduction

A deductive argument has been defined by MacCormick as:

"an argument which purports to show that one proposition, the conclusion of the argument, is implied by some other proposition or propositions, the 'premisses' of the argument. A deductive argument is valid if, whatever may be the content of the premisses and the conclusion, its form is such that its premisses do in fact imply (or entail) the conclusion".

[MacCormick, p21, 1978]

In essence, a deductive argument is composed of two parts, between which there is an implied consequence such that if one part (the antecedent) is held to be true, then the other part (the consequent) is caused to come about (i.e. loosely, "true"). The actual contents of the parts are unimportant; the only requirement is that a truth value can be returned by the antecedent. Thus the general inferential format of a deductive argument is:

if antecedent is true then consequent

or more symbolically (but not more formally):

p then q.

Given a deductive argument of this form, it is self contradictory for anyone to claim that if "a" then "b" is not implied. As an example, if the antecedent (or proposition) is:

"his name is Joe"

and the consequent is:

"the person named in the antecedent is male"

then assuming the antecedent to be true (and assuming that we are detailing intra-systematic scope of the referents - as we shall do constantly from now), we must accept that Joe is male.

Note that implication is one of the the most argued about aspects of logic, though. For example, deduction is often modelled by A B; but this is more correctly read, some claim, as "if B then A" [De Long, 1970, p98-99]. We return to this in the final section of this chapter.

#### 4.8.2 Production Systems

Such an antecedent/consequent pair is the central element of a production system [Post,1943]. In a production system the antecedent and consequent pair are described as production rules, productions or, simply, rules. They are generally but not always, held as a vertical list:

IF a THEN b

IF c THEN d

IF e THEN f

.. ..

IF x THEN y

Such rules are the rules which govern behaviour within the system. Thus where the formalism has been utilized in combinatorial work the productions represent graphical operations, where they appear as Chomsky's 'rewrite rules' they govern linguistic or grammatical operations. One of the most important aspects of productions are that,

when applied, they produce a change in the environment of the system (hence the name production) - for instance a higher level phrase is produced with Chomsky's application of a rewrite rule.

The application of productions is carried out in a deterministic manner - the productions are said to be "interpreted". While there are a variety of particular ways in which productions can be interpreted, there is one general axiom - only one production can be applied at any one time. This implies a serial ordering of rule application. Usually the ordering is from top to bottom; each production is tested to ascertain whether its antecedent is true - if it is true then that production is triggered, if not triggered the following rule is tested. Triggering a rule means effecting the consequent - which can be considered similar to proving the consequence "true" - and then returning to the top of the list to restart the testing operation.

At this point no interconnection has been shown between antecedent and consequent apart from a causal link - if the antecedent is true it causes the consequent to come into effect. However, it is usual for the consequent of one rule to be an element in the antecedent of another rule. Thus the productions in this simple example:

IF a and b THEN c

IF f and a THEN g

IF g THEN d

If c THEN f

IF a and d THEN e

are perfectly possible. In order to bring into effect e, it is essential that a and b are, within the system, initially true. If only one or neither of a and b are true, then c cannot be effected. If c cannot be effected then neither can (by way of f and g) d - which is a necessary element of the production whose consequent is e. If a and b



are true, then the first production can be triggered; this effects c. The second production cannot because we do not know that f is true. Effectively, the consequent of the third production is debarred from effecting because g cannot be proven. Testing the next production, because we know that c is true (it was effected by the first production) f is effected. Since after each occasion that a consequent is effected we return to the top of the list again, we then scan down for an untriggered production whose consequent we can effect, etc.. The procedure continues in this manner until e is effected.

This simple example uses only 5 rules, works towards only one highest possible effect (i.e. e), and does not request any input from outwith the system apart from the initial starting values. In programs using this technique, the minimum number of rules is usually about 150, various eventual conclusions might be achieved (e.g. with Chomsky's rewrite rules, a large number of different, grammatically correct sentences can be produced), and much more extra-systematic interaction - and request for data - is necessary.

A skeletal outline of the above process is:

IF a and b THEN e

but this does not indicate the "reasoning"; while technically correct it does not show the flavour of the reasoning steps from initial testing through to conclusion:

(a and b) then c then f then (f and a) then g then  
d then (d and a) then e

One important point not fully illustrated by this small example is that an object produced by a production can be said to be defined by it. Thus in:

IF a and b THEN c

c can be said to be defined by a and b. If two productions of the form:

IF a and b THEN c

IF d and e THEN c

c can be said to be defined by a and b and/or d and e. These are described as definitional rules.

This production system formalism is an example of a second order formal logical system [Enc. of Phil.,1967,8,p61/74]. A formal logical system is a simple mathematical device. It is a system for explicating the notion of consequence. This consequentialist process is shown by the use of certain aspects of a formal language:

- (1) a set of transformation rules.
- (2) a set of axioms concerning application of 1.
- (3) formation rules for 1.
- (4) a list of symbols transformable by 1.

Thus it is possible to see from this definition that a production system is a formal system:

- (i) its procedures are its transformation rules.
- (ii) its manner of interpretation is its application axioms.
- (iii) its productions are of a set format, i.e. IF p THEN q.
- (iv) those symbols specified in its productions constitute  
its list.

While outwardly it may seem that a legal system is similar to a formal system, for example that it consists of a body of rules which must be interpreted etc, it is not our intention to prove, or our belief that, law is a formal system.

#### 4.8.3 A Legal Example

While the original legislative domain used in our research was Welfare Rights Legislation (used as test data, we originally considered, because of its cohesive and "easy" - rather than "hard" - predictive nature) it is useful to discuss another field of legislation at this time. The usefulness arises because a current controversy exists over the interpretation of a particular judicial judgement as either "deductive" or "inductive". The case is that of Daniels and Daniels v R. White & Sons and Tarbard ([1938] 4 All E.R. 258). MacCormick provided a synopsis of the case:

"The facts of the matter were as follows. Mr Daniels went to a pub, and there bought a bottle of lemonade (R. White's lemonade) and a jug of beer. These he took home, and there drank some of the lemonade himself and gave a glass of it to his wife, which she drank. They both experienced burning sensations and became ill. The cause of their sickness was subsequently established as being the fact that the lemonade which they had consumed was heavily contaminated with carbolic acid. Examination of the remaining contents of the lemonade bottle showed the lemonade to contain a large admixture of carbolic acid.

The plaintiffs, Mr. and Mrs. Daniels, subsequently sued the manufacturer of the lemonade and the publican who sold it to them for damages in compensation for their illness, treatment expenses, and loss of earnings when ill. The defendant manufacturer was absolved from liability ... ; the defendant publican was held liable and ordered to pay them damages."

[MacCormick, 1978, p19/20]

The reasoning process behind (rather than the reason for) such a judgement is the core of contention - was the decision arrived at "logically" (i.e. deductively) or was it arrived at by "classification" (i.e. inductively). Using a corpus of productions it is possible to illustrate a deductive process which might (and I deal with this "might" later) have caused the judgement to be arrived at. Such a corpus of rules does not exist solely within legislation or solely within case precedents. Rather it must be extracted from legislation, precedent and expertise and tested as to its validity.

The corpus (or knowledge base in IKBS terms) here consists of only a few rules; they are used only to exhibit relevant points since a fuller description of the mechanical handling is available elsewhere (Chapter 6).

However, the rules are:

1. IF <a sale has taken place>  
    <goods requested by description>  
    <seller deals in described goods>  
    THEN  
        <sale by description>
2. IF <sale by description>  
    <defects cannot be observed by examination>  
    THEN  
        <goods are not of merchantable quality>
3. IF <object acceptable after examination>  
    <object fit for purpose bought>  
    THEN  
        <goods are of merchantable quality>
4. IF <person has power over goods>  
    THEN  
        <person has power to sell property>

```

5. IF  <person is a Sheriff>
      <goods seized under writ of execution>

      THEN

      <person has power over goods>

6. IF  <a contract of sale has been made>

      THEN

      <a sale has taken place>

7. IF  <object transferred from seller to buyer>
      <person has power to sell property>
      <transfer completed for monetary consideration>

      THEN

      <a contract of sale has been made>

8. IF  <a written contract prepared>
      <transfer completed for monetary consideration>
      <transfer to take effect at future time>

      THEN

      <a contract of agreement to sell has been made>

9. IF  <goods are not of merchantable quality>
      <a sale has taken place>

      THEN

      <seller liable for damages to buyer>

```

At first sight, and especially to those with contact only with first order logical systems, the above may appear to have little relationship to "logic". However, they should be seen as rules where the positions taken by variables contain strings of meaningless marks between "<" and ">" symbols, and that:

```
IF <A><B> THEN <C>
```

means

```
IF (<A> AND <B>) THEN <C>
```

It is possible, of course, to utilize other logical connectives than

"AND" - e.g. "OR" etc. When initially deciding on the formalism we chose to put more effort into simplifying the rules before incorporating them within the system because I believe that this will enable the corpus to be kept up to date more easily by non-computer scientists.

The validity of having a logical system which deals with marks was noted first by C.I.Lewis and was described as the 'heterodox view of mathematics and logic':

"A mathematical system is any set of strings of recognizable marks in which some of the strings are taken initially and the remainder derived from them by operations performed according to rules which are independent of any meaning assigned to the marks. That a system should consist of marks instead of sounds or odors is immaterial, but it is convenient to discuss mathematics as written. The string-like arrangement is due simply to our habits of notation. And there is no theoretical reason why a single mark may not, in some cases, be recognized as a "string".

The distinctive feature of this definition lies in the fact that it regards mathematics as dealing, not with certain denoted things - numbers, triangles, etc. - nor with certain symbolized "concepts" or "meanings", but solely with recognizable marks, and dealing with them in such wise that it is wholly independent of any question as to what the marks represent.

The question of logical meaning, like the question of empirical denotation, may be regarded as one possible application and not of anything internal to the system itself."

[Lewis, p355/356, 1918]

Therefore, viewing these marks as non-meaningful within the system itself does not preclude assigning an extra-systematic value to them.

Such a value is, in effect, built into the basic propositions or axioms of the system and, as required by this non-modal system, necessitates only the ability to assume the value "true" or the value "false". This approach is advantageous because it separates the meaning of the marks from the deductive process, the latter relying upon only the similarity of the marks, e.g. <sale by description> has the same meaning as <sale by description> but not <a contract of sale has been made>. This approach allows the marks (or "conditions" or "states" if preferred) to be semantic tokens to which annotations, references to law reports and legislation etc. can be added to help the person interacting with the system determine whether "<...>" is "true" or "false". The only logical requirement is that for symbols to be equivalent they should have the same marks and annotations. We simplify. In fact there is no need to be quite so pedantic - what is necessary, though, is that the symbols should be conceptually identical; which is rather different from having identical marks. Thus "monday morning" is conceptually identical (in most cases) with "monday between 00.00.01 hrs and 11.59.59 hrs".

The models interprets its production rules in the following manner. Rule 1 is tested first. The first condition of the antecedent is <a sale has taken place>, and the system must determine the truth or falsity of this. How can it be done? There are three possible strategies which can be applied:

(i) the system might already have determined that the condition is true. In this case the strategy is simply to search a list of known facts; if the condition is found in the list it is "true", else attempt (ii).  
(ii) by "backchaining", as it is known in computer science. Thus an incidence of the condition appearing as consequent of another rule (as condition 1 of Rule 1 in Rule 6) allows the system to make use of the definitional character of the rules - i.e. it can backchain through the antecedents of the "defining rule" to determine whether its consequent is effected, and therefore whether the condition in the first rule is

true. Whenever a condition is proven true, it is added to the list of conditions which (i) can access. If no incident of the condition as consequent is found, or if the backchaining strategy is unsuccessful, then strategy (iii) is applied. (iii) The user can be asked whether the condition is true or false; which is equivalent to posing the statement "in this case Socrates" in the modus ponens.

In the following, for shorthand purposes each strategy will be referenced by (i), (ii) or (iii) as appropriate, and R1 represents Rule 1, C1 represents condition 1 etc. The example has been chosen more to illustrate the definitional (i.e. backchaining) aspect of interpretation. Interpretation is of the form:

- 1 apply (i) to R1C1 - not known
- 2 apply (ii) to R1C1 - match found at Rule 6
- 3 apply (i) - to R6C1 not known
- 4 apply (ii) to R6C1 - match found at Rule 7
- 5 apply (i) to R7C1 - not known
- 6 apply (ii) to R7C1 - no match found
- 7 apply (iii) to R7C1

at this point, the system must ask the user whether <object transferred from seller to buyer> is true. If so (as we shall assume) then the next condition is tested:

- 9 apply (i) to R7C2 - not known
- 10 apply (ii) to R7C2 - match found at Rule 4
- 11 apply (i) to R4C1 - not known
- 12 apply (ii) to R4C1 - match found at Rule 5
- 13 apply (i) to R5C1 - not known
- 14 apply (ii) R5C1 - no match found
- 15 apply (iii) to R5C1:

at this point the user replies 'false' to the system's request for the



truth or falsity of <person is a Sheriff>. Therefore Rule 5 cannot be triggered, and the interpreter returns to trying to trigger Rule 4, continuing the process started at step 12:

16 apply (ii) to R4C1 - no match found

17 apply (iii) to R4C1:

asking the user once again. This process continues, building up the list of known conditions, and working through the rules in the system. While tedious to follow here, such a repetitive procedure (or algorithm) is just how a computer operates.

During this interpretation, the user may be asked the truth of, for example, <person has power over goods>. It may be that other rules govern the definition of this concept (as would be necessary in any system which might be "expert" in the Sale of Goods legislation) but none exist in the example's corpus - therefore the user has recourse to only 'common sense'. In a realistic system these cases, where the user has no aid from other defining rules, might be at a higher conceptual level for those with experience of the legislation, and at a very low conceptual level for those with no experience of the law at all. This necessity to reduce all concepts to un-confusable english words intelligible by persons with no experience of the law is one reason why we are sceptical of claims to produce "mechanized justice".

Concluding, there are several points which this example exhibits:

(i) the "truth" or "falsity" of the individual antecedents of each rule are eventually decided by extra-systematic interaction, in the same way that we must interpret the syllogism.

(ii) as in all deductive systems, only that which is already within the system can be concluded from the premisses (i.e. the system does not make new legal decisions - it only applies past ones).

(iii) the process is inferential, i.e. deductive.

(iv) a deductive process can allow rules to be used in a definitional manner.

In the next section, we shall see how closely this model might accord with legal reasoning as it occurs in court.

#### 4.9 THE AFFINITY OF THE MODEL TO ACTUAL LEGAL REASONING.

##### 4.9.1 Introduction.

There are various problems which arise when an attempt is made to compare a model with that which it models. In some circumstances these problems are less important than in others. For example, when comparing a Treasury model with the financial state of the country, the comparison is between a model and abstract economic concepts (e.g. "value of international trade") and no attempt is made to ascertain whether the model operates in the same way as its object of study. When, however, one of the principle goals of the model is to help decide just how its object of study operates, the problems are akin to trying to see into a black box - much can be proposed about the internal operation, but little can be proven.

Another general problem is that of deciding how far any future extension of the model might be expanded - thus when does a model become, if not reality, at least a precise explanatory copy of reality.

The first is important in the context of legal reasoning because we are attempting to model what happens within a judge's person (or "mind"). The second is important because it deals with the possibility of "mechanized justice".

##### 4.9.2 Psychologicalism

Any theory of legal reasoning must, at some point, involve notions of what passed through the judge's mind when the decision was being made. There are problems, though, in deciding just how this mental process might be observed. An example of the difficulty an individual has in expressing his thought processes, is found in the current research in computer science which attempts to incorporate an expert's knowledge about his domain within a computer program. This research has discovered that frequently an expert cannot express just why, in certain situations, he applies one piece of knowledge rather than another; or sometimes have difficulty in enunciating knowledge which he has applied.

One possible way to circumvent mind reading is to use the written judge's decision. MacCormick uses these decisions to illustrate the decision process of the judge, claiming that the decision was arrived at logically. But Wilson refutes that the judge's reasoning had a deductive nature:

"... there seems to be, in this particular case, no good grounds to believe that Lewis J reasoned in these three allegedly 'syllogistic' stages ..."

[Wilson, p280, 1982]

The sceptic, of course, might helpfully agree with Wilson and posit that no sane judge is to be expected to present an argument in a "non-logical" way: whether the reasoning was logical differs from whether the presentation is logical.

Surprisingly, even the production system formalism has been seen by cognitive psychologists as possibly how the "human program" operates as noted above by [Newell, p806, 1972].

It seems that such propositions concerning actual mental actions can never be proven; the debate therefore seems empty. Therefore, I will make no claims for judges actually thinking deductively - only, in the next part, that they it is possible to model them as exhibiting deductive behaviour. The usefulness of the model lies in the analytical insights it helps provide: these insights we return to later.

#### 4.9.3 Deduction Or Induction

Wilson puts forward the inductive argument as:

"Thus the process whereby the judge decides that the object, action or transaction facing him is to be classified under a particular legal category - such as 'contract' or 'delict' will constitute the central element in his reasoning. In this process the judge performs the mental operations of 'identification' and 'classification': identification of the empirical data - be it a present object or evidence of past actions - and its classification under a relevant legal figure whence legal consequences will follow. This crucial operation is not deductive. In deduction, the data are presented to the reasoner already identified and classified."

[Wilson, p278/279, 1982]

We shall omit one of the common arguments against this perspective - i.e. that the judge has a case presented to him, he does not 'classify' and 'identify' like some legal ghost in the machine - and concentrate on the proposition that induction may be viewed as a process of guesswork and deductive justification as proposed by Russell. Such a view would see the judge taking a tentative position of what "class" a certain object or occurrence might be contained within, and then testing deductively whether there is some chain or reasoning which might support

such a position. If not, another position is taken, else that first position is accepted as justified.

Such a possibility has been ignored by researchers in legal reasoning. For example Wasserstrom believes that this process of classification is not deductive even though he does see legal reasoning as primarily deductive:

"... it would still be true that "logic does not prescribe interpretations of terms; it dictates neither the stupid nor intelligent interpretation of any expression". One cannot appeal to the canons of logic to decide whether a given classification is necessarily the correct one"

[Wasserstrom, p32/33, 1961].

Russell's argument is expressed succinctly as:

"You observe that  $1 + 3 = 2$  ,  $1 + 3 + 5 = 3$  ,  $1 + 3 + 5 + 7 = 4$  , and you say to yourself: "in some cases, the sum of the first  $n$  odd numbers is  $n$  ; perhaps this is true in all cases." As soon as this hypothesis has occurred to you, it is easy to prove that it is correct. In empirical material, a complete enumeration may sometimes be possible."

[Russell, p89, 1980]

Just as a mathematician is much more likely to intuitively notice the connection between sums of odd numbers and the square of the number of those, it is surely reasonable to expect that a judge, with training and expertise in the law, is capable of formulating questions of the sort, "Given that many persons, having been found without authority on private property carrying bags marked "swag", have been found guilty of theft;

perhaps this present person should be so found as well." Such an inductive (or perhaps "intuitive") proposition needs recourse to some sort of justification for its belief. The justification will probably be both recourse to the usage of the term "without authority", probably evidence to the contents of the sack etc. But there must be justification; and that justification must be legally relevant.

Similarly, the assumption that an incidence of "sale by description" has occurred also requires justification. How might this process of justification be arrived at?

One possibility is that some process similar to that occurring within the interpreting section of the model is being put into effect - i.e. that inferences (i.e. a deductive process is occurring) are being made from the basic propositions of the system, and that those propositions (i.e. legal rules) are being used in a definitional manner.

#### 4.9.4 Law As A Formal System

The notion that law is formal system might appeal to some within computer science - implying the possibility of building automated purveyors of justice. But to see whether such a vision is possible, it is necessary to determine whether the law is, or could be, a formal system, since as mentioned earlier this explication is necessary in order that it can be computerized.

Earlier, we noted that our research was not directed towards proving that the law was a formal system - i.e. that is consists of an axiomatic basis and an inflexible procedure for deducing conclusions. If the law were a formal system the relationship between it and life would not be in doubt - the law would stand in a directive relationship to the world; that is a one way relationship. But this is obviously not the case. It stands, not solely in a directive relationship to the world (e.g. as a command theory would have us believe), but in a two

way relationship. Thus the law (and by that we mean the application of the law) can be affected by external influence. A formal system by definition, some claim, cannot be so affected - its axioms of interpretation are precise and constant over time; and cannot be affected by such considerations as might influence the decision of a judge. (We disagree that these systems are "set in concrete", but agree that it is difficult to have them affected by ad-hoc judicial behaviour.) For example how might a formal system cope with MacCormick's contention that it is possible that Mrs. McTavish was granted her a divorce

"just because she has a ravishingly pert retrousse nose"

[MacCormick, p15, 1978]

Other work by Griffiths [1977] has shown that innate conservatism of the judiciary has effect upon judicial decision. Griffith denotes the cause of this application of conservatism as "judicial creativity" and points to its powerful effect:

"I have said that judges look to what they regard as the interests of the whole society. That, in itself, makes political assumptions of some magnitude. It has long been argued that the concept of the whole society suggests a homogeneity of interest amongst the different classes within that society which is false. And that this concept is used to persuade the governed that not the Government but the 'State' is the highest organization and transcends conflicts in society. It is a short step to say that it is the State which makes the laws, thus enabling those in political power to promote their own interests in the name of the whole abstracted society."

[Griffiths, p204, 1977]

Reed C. Lawlor [1980] has also provided evidence of a strong personal (that is individual) element in application of stare decisis. If the law is a formal system, blatantly these occurrences would be unobservable; the fact that they are observable illuminates only the fact that the law is not a formal system.

On the other hand if such "alien elements" could be eliminated from the law, would the law be akin to a formal system? We suspect not; for a formal system can only treat its variables as marks upon paper, and when such marks are transliterated into english text the semantic problems of interpretation still arise. The application of law is a social activity, necessarily dependant upon the use of language, not something which can occur outside of social activity. Thus if we ever do provide computerized justice, there must still be people capable of interpreting the computer's decisions; either that or the computer must have the ability to precisely specify every legal concept by application of simple english words which can never be mis-understood. But such words are, of course, in short supply.

A third point is that a formal system can only conclude from that which is already within its propositions (i.e. its rules if it is a production system formalism). Any attempt to produce a computerized system which can deal with every eventuality is, I feel, difficult to accept as feasible. Judges, with their abilities to reason by "analogy" and "principle" are quite safe from computerization.

#### 4.9.5 Conclusion: The Myth Of Logical Compulsion

There has been an implicit position which we have hinted at throughout this chapter, a position is far from usual in discussion of logic. Our position is relativist, viewing logical systems as tools while discarding any notion of extra-systematic validity they might have. This means that we see logic as useful, but not something which offers



us a stable platform to argue from; put simply, logic is too problematical to be any platform except, perhaps, a leaky raft in the North Sea. We have called into question logic's relationship to the law by noting the complexities of logic (later we shall do so by noting the complexities of law), but also our argument has been against the notion of logical compulsion - that logic compels us to do anything. In this concluding section we bring out this aspect of logic. Our argument is, of course, that logic cannot be a foundation for law.

We have dealt with MacCormick's belief in the special validity of logical form: we can see that he believes, for reasons of the desire to be non-contradictory, that a deductive argument has a special force:

"A deductive argument is valid if, whatever may be the content of the premisses and the conclusion, its form is such that its premisses do in fact imply (or entail) the conclusion. By that is meant that it would be self-contradictory for anyone to assert the premisses and at the same time to deny the conclusion."

[MacCormick, p21/22, 1978]

It is the abstraction of logical form which is important to MacCormick since that abstraction can offer itself as a guide to any contradiction in an argument. Such a belief is one which can be termed, the belief of logical compulsion, since it allows us no respite from its compulsiveness; it would be as MacCormick states, "self-contradictory for anyone to assert" against its compulsiveness. The belief in this compulsion is a powerful tool (if valid) to have in the toolbox of the philosopher (and the judiciary) because it allows the philosopher to resort to an external and stable reasoning mechanism; it is a tool which is central to MacCormick's position. We argue that it is a position which is essentially Platonist, and not supported by empirical and historical evidence.

Maurice Kline [Kline, p4, 1980] has stated that, "The most fertile source of insight is hindsight", and indeed, most of the arguments and facts against the perceived power of logical compulsion are historical and empirical, viewing logic as a developing element of mathematics. It is just this empirical examination which will provide important evidence in our argument against logical compulsion and pure deductive reason, just as empirical evidence provides the best argument against a belief in clear legal rules (Chapter 9).

A primary element of the positivist legal theorist is that there is a certainty associated with logic (for example, MacCormick's quote above about the necessity to accept the power of argument which uses logical compulsion - we cannot argue, we can do nothing but accept the certainty of the logically formulated argument). Unfortunately, examination of the conceptions of logic and mathematics illustrate that there are problematic aspects about seeing certainty as one of their attributes.

One instance of the development of logic and mathematics is the "loss of certainty" which has occurred in the 20th century, demonstrated by these two quotations from Bertrand Russell in two periods from his life, the first from that early period when logic (or, more particularly to Russell, mathematics to which logic was inextricably bound) was seen as a fount of certainty:

"Of such scepticism mathematics is a perpetual reproof; for its edifice of truths stands unshakable and inexpugnable to all the weapons of doubting scepticism."

[Russell, 1903]

and the second, when that certainty had passed away like sand through the fingers, looking back on his attempts to give arithmetic a sound logical foundation in the Principia Mathematica project:

"I wanted certainty in the kind of way in which people want

religious faith. I thought that certainty is more likely to be found in mathematics than elsewhere. But I discovered that many mathematical demonstrations, which my teachers expected me to accept, were full of fallacies, and that, if certainty were indeed discoverable in mathematics, it would be in a new field of mathematics, with more solid foundations than those that had hitherto been thought secure. But as the work proceeded, I was continually reminded of the fable about the elephant and the tortoise. Having constructed an elephant upon which the mathematical world could rest, I found the elephant tottering, and proceeded to construct a tortoise to keep the elephant from falling. But the tortoise was no more secure than the elephant, and after some twenty years of very arduous toil, I came to the conclusion there was nothing more that I could do in the way of making mathematical knowledge indubitable."

[Russell, p53, 1956]

The importance which Russell saw in logic, in the early period, was that it acted as a barricade against universal scepticism. When in mathematical logic (which is concerned, for example, with problems such as infinite sets and other set theoretic concepts and suchlike - not the simple problems of modus ponens) certain problems (paradoxes) arose, Russell felt the need for a sound philosophical view of logic and mathematics. He wrote:

"I wrote to Frege about it (1) [the logical problems], who replied that arithmetic was tottering. He was so disturbed by this that he gave up the attempt to deduce arithmetic from logic, to which,

---

(1) The story arises of how just as Frege was going to press with the second volume of his "Fundamental Laws of Mathematics", Russell's letter arrived. At the close of the volume Frege remarked, "A scientist can hardly meet with anything more undesirable than to have the foundation give way just as the work is finished. A letter from Mr. Bertrand

/Russell

until then, his life had been mainly devoted. For my part the trouble lay in logic rather than in mathematics and that it was logic which would have to be reformed."

[Russell, p37, 2nd Introduction to Principia, Cambridge Paperback]

The need for this reformation was not because the paradoxes were of any importance to mathematicians; they were getting on with the job just as mathematicians always do (compare with the situation where lawyers are rarely concerned with jurisprudential problems). The need, which Russell saw, was philosophical - how to ensure that sceptics could not pounce upon the contradictions and "problems" of mathematics - the problem was, of course, that logic might not be a stable reasoning form.

Thus, logic was used by Russell (and most other philosophical mathematicians of the time) as part of his philosophical structure; it was not simply a tool to be used in mathematics - it, or a given view of it, was considered essential by Russell to support his view of the world. We suggest that researchers in legal logic have paid too little attention to hindsight and have taken, without proper consideration, that same view of the world as Russell first took - i.e. that it is logically structured.

This world view has led researchers into logic and the law to usually consider the prescriptive aspects of logic to be as important as the technical aspects. Take, for example, Ilmar Temmelo:

"In legal reasoning formal consistency of thought is an end constantly pursued, even though it is not always achieved. ...

---

Russell put me in this position at the moment the work was nearly through the press." Russell also wrote [Russell, p76/7, 1959], "Philosophers and mathematicians reacted in various different ways to this situation. Poincare, who disliked mathematical logic and had accused it of being sterile, it begets contradiction, exclaimed with glee, 'it is no longer sterile, it begets contradiction'. This was all very well, but it did nothing towards the solution of the problem. Some other mathematicians, who disapproved of Georg Cantor, adopted the March Hare's solution: 'I'm tired of this. Let's change the subject'. This, also, appeared to me inadequate."

Logic is significant for the lawyer in that it helps him to present his reasoning in a well-organised, lucid, and cogent manner. The actual presentation of his reasoning must be taken into account, of course, the addressees of his train of thought and must be adjusted, by employing appropriate informal ways of expression, to their intellectual background and habits of thought. But this train of thought is more likely to be sound if it is established in awareness of the formal requirements of self-consistent reasoning. The same awareness is important for legal draftsment. Antimonies, gaps, ambiguities, and vaguenesses in law can be avoided if the draftsmen are conversant with principles and methods of logic. ... Logical rigour means intellectual integrity and is thus an important ethical requirement in the application of the law. To blame logic for shortcomings in the administration of the law is very much the same as to blame honesty for evils in the world."

[Tammelo, p132/133, 1969]

Tammelo believes, then, that self-consistent argument and reasoning will win through (just as long as the intellectual background of its recipient is taken into account). The vision which Tammelo conjures up is of the rational lawyer arguing in an irrational courtroom - if he keeps his head, then not all is lost. The second point which comes out is that the clarity of legislation is of fundamental importance, and that logic can be used to aid correct interpretation of that legislation. Thirdly, that ethical integrity is achieved by a technical application of reasoning.

Tammelo's view has much in common with a claim made 381 years previously that other researcher into logic and the law, Abraham Fraunce. We might conclude from these researchers that the history of legal logic as the explication of correct reasoning has a long and consistent history. Unfortunately, that is not the case - Tammello's logic is that of

Aristotle and Fraunce's is that of Ramus. There is a substantial - some might say unbridgeable - difference between the two logics.

Pierre Ramus, the deviser of Ramist logic, was a Frenchman whose work epitomized the medieval criticism of the scholastic teaching of Aristotle. It was the teaching of the Organon to young boys in medieval universities in a manner similar to the teaching of multiplication tables (in the days before the introduction of the electronic calculator) which gave rise to much cynicism about the power of, for example, *modus ponens*. Ong [Ong, 1958] has pointed out that the view of Aristotelean logic which is the object of that medieval cynicism is not the Aristotelean logic of Aristotle (1). It was the early part of Peter of Spain's *Summulae logicales* which was used as the outline of Aristotelean logic, a text book used by teenage university students to learn the arts of dialectic. It opens with the statement, "Dialectic is the art of arts and the science of sciences, possessing the way to the principles of all curriculum subjects."

Ramus took this general spirit of universal applicability as the starting point for a logical method which could be used in all the arts; it was a means of overcoming the dryness of the scholastic logic and applying it to the real world, rather like taking the rote of multiplication and using it in a computer model of rice harvesting in the third world; an activity so popular in today's primary school education. Ramus's lectures, we hear, were popular events attended by huge crowds, the lectures peppered with verbal assaults on the material and authors taught by his fellow professors. The cries of "Holy Jupiter!" and "Good God!" were complemented by "spectacular declamation

---

(1) Misrepresentation being a common enough factor even today in jurisprudence - see [Moles, 1985] on Hart's interpretation of John Austin. Also we can quote Durkheim's introduction to the 2nd edition of his "The Rules ..." where he stated, "In fact, certain opinions were imputed to us that we had not put forward, on the pretence that 'they were in conformity with our principles' " [Durkheim, 1982]. The situation must have been even worse when the recipients of the misinterpretations were teenage students with little opportunity to research the original writers themselves.

and gesture." In comparison with the teaching of the syllogism at the time, it must have seemed a wonderful improvement in educational strategy, rather like the overthrow of the three R's by "the project" which occurred in this country in the 1960's.

However, Ramist logic differs from what we might term logic today. It accords much more with certain tendencies in General Systems Theory to use diagrammatic representations as descriptions of the world. Ramist used diagrams to represent dichotomies as systems theorists will use diagrams to represent "systems".

The general Ramist method was to examine a logical class and divide it into two parts. This has been pointed out as the method used in his discourse on grammar; it was natural therefore to externalise the process and claim the:

"natural method is essentially the method of two dichotomies - of proceeding always to separate a logical class into two subclasses opposed to each other by contradiction, and to separate the subclasses and sub-subclasses in the same way, until the entire structure of any science resembled a severely geometrical pattern of bifurcations."

[Howell, p162, 1956]

As an example of this method, we might see how Fraunce took Ramist logic and applied it to an outline of "The Lawyer's Logicke" where each class is divided into two sub-classes etc. The same procedure was given by Fraunce for the case of the Earl of Northumberland (although he notes, "For the Earle of Northumberlands case, I have rather shewed what is done by Maister Plowden, than what might have beene done by a better Logician.") and "Stanfords crown pleaes".

But before we get excited about the usefulness of Ramist logic, it is helpful to note what Ong has written of this logic:

"As a popularized or residual class logic, the Ramist dialectic manifests a quantification system which is almost certainly the most recklessly applied one that the world has ever seen."

[Ong, p203, 1958]

Obviously, then, it is difficult to make any claim for similarities between Fraunce's logic and that of Tammelo (or MacCormick). It seems, therefore, that while initially we might think that legal logic has some consistent history, on examination, we find that this is not so; the consistency might arise from the word "logic" but most certainly does not arise from the "method of logic".

It is always possible to say, as some do, that Ramist logic "is not logic". Such a position is, as Haack notes, monist; but does a monist position stand up to empirical evidence? Can this strict demarcation of the validity of logical systems be applied to "traditional" logic? We would argue, again by looking at empirical evidence, that this demonstrably cannot be done; even in what might be called traditional logic there are a variety of ways of interpreting the notion of implication.

MacCormick interprets "p implies q" as "if p then q", whilst other logicians (for example, Russell in Principia) treated "p implies q" as equivalent to "not(p and not q)". While MacCormick's implication is causal, in that it demonstrates how the consequent "q" can be concluded, Russell's is not - it is based upon the use of negation, not "if .. then". Russell was actually unhappy about expressing implication in any natural terms at all and only used his expression as the most convenient form (in that it fitted in notationally with the Principia). Unfortunately, this implication leads to intuitive problems - first that



a false proposition implies any proposition and second that a true proposition is implied by any other proposition. These problems (basically that Russell's implication was "intuitively incorrect") led C.I. Lewis to feel the need for "strict implication". In fact, looking at the problems which attempts to define implication have come across, it is almost the case that there are as many interpretations of implication as there are logicians. And the problem is not simply a modern one. We can quote Mates on the problem implication [referred to in the extract as truth-conditions of conditional propositions] caused to the Stoics:

"It seems that questions of logic were taken very seriously in ancient times. When Diodorus Cronus was unable immediately to solve a logical puzzle proposed to him at a royal banquet in Alexandria, he died in despair. Philetas of Cos, another logician, was a victim of the famous antimony of The Liar, as we know from his epitath:

Philetas of Cos am I

"Twas The Liar who made me die,

And the bad nights caused thereby."

Likewise the problem of the truth-conditions of conditional propositions, although it apparently caused no fatalities, inspired so much discussion in Alexandria that Callimachus reports, "Even the crows on the roof tops are cawing about the question which conditionals are true."

[Mates, p43, 1961]

Other examples of the different views of logical systems arise when we leave traditional logic. In non-traditional logic we can perfectly easily do away with the law of the excluded middle (which says that only a value of true or false can be handled) and we can speak of a variety

of possible values. Modal logics also each have their meaning described in a different manner - the rules of inference within the systems differ. It is, in fact, one of Quine's attacks upon the use of modal logic (he is a virulent opponent of them) that the interpretation of modal logics is fraught with severe difficulties - Quine noted several different ways of using the concept of "necessarily", for instance.

Having looked, then, at some of the problems of logic, it is hard to accept MacCormick's simplistic view of the notion of logical compulsion as valid - that logic solves without further consideration the question of self-contradiction. But we can go further.

The logic which MacCormick utilizes is that of Aristotle, a logic which arose from the geometric systematization of Euclid. Euclid's task was to find a universal set of axioms - what we might call assumptions or first principles as Aristotle termed them - which would give a rational structure to geometry. Such a task was of prime importance at a time when geometrical constructions were of religious and philosophical importance(1). Most people have had some contact with Euclid's elements - the definition of a point as "that which has no parts" - during their school education, and have learned that Euclid's axiomatization is only relevant in non-curved space (in curved space non-Euclidean geometry takes effect). One point about Euclid's axiomatization is that it is essentially static and offers no indication of change or expansion of the a priori knowledge. Euclid's method was seized and applied by philosophers as the epitome of rational intelligence. Aristotle was one such, and as he states in the Posterior Analytics, he believed that the correct manner to work in every science was to find the fundamental axioms or first principles and build upon these. The building was to be

---

(1) The Pythagorean school of geometers would not allow instruments to be used in building their constructions, because of the religious impurity that would be imported into the study. Mind you, the Pythagoreans were celibates who performed ceremonial purgations and would not walk on high roads or sit on quart measures, so the revulsion towards geometrical instruments is hardly the oddest of their behaviours.

carried out by either inductive or deductive reasoning, whose form he expressed in the Organon.

Thus logic arose from the formalization of geometrical reason. It was accepted because of its inherent usefulness - it was intuitively useful as a representation of correct thinking.

But there is substantial evidence to believe that the geometrical reasoning processes from which logic was derived are not all that they might seem at first sight. Geometrical reason, this evidence suggests, is a reason arrived at by hindsight; the arriving at the definition of geometrical structures is a process more akin to negotiation of a social kind. Perhaps the most illuminating study of this was carried out by Lakatos [Lakatos, 1976] in the analysis of the development of the concept of polyhedron - as definitions of the object were provided, other mathematicians would try to include other objects within the definition or would discover objects which could be included within the definition but which were not classed by other researchers as that first object.

There is no need here to repeat Lakatos's study, but we do need to note the points that he makes about attempts to formalize geometry and mathematics - by formalizing, he means claiming that mathematicians can work from basic axioms (or premisses) and build up and extend bodies of mathematical knowledge. The extract which we use is of particular importance for MacCormick, as a proponent of the deductive approach to legal reasoning:

"The student of mathematics is obliged, according to the Euclidean ritual, to attend this conjuring act without asking questions either about the background or about how this sleight-of-hand is performed. If the student by chance discovers that some of the unseemly definitions are proof-generated, if he simply wonders how these definitions, lemmas and the theorem can possibly precede the

proof, the conjuror will ostracize him for this display of mathematical immaturity. (Some textbooks claim that they do not expect the reader to have any previous knowledge, only a certain mathematical maturity. This frequently means that they expect the reader to be endowed with the "ability" to take a Euclidean argument without any unnatural interest in the problem-background, in the heuristic behind the argument.)

In deductivist style, all propositions are true and all inferences valid. Mathematics is presented as an ever-increasing set of eternal, immutable truths. Counterexamples, refutations, criticism cannot possibly enter. An authoritarian air is secured for the subject by beginning with disguised monster-barring and proof-generated definitions and with the fully-fledged theorem, and by suppressing the primitive conjecture, the refutations, and the criticism of the proof. Deductivist style hides the struggle, hides the adventure. The whole story vanishes, the successive tentative formulations of the theorem in the course of the proof-procedure are doomed to oblivion while the end result is exalted into infallibility."

[Lakatos, p142, 1976]

Lakatos points out that we should be wary of the deductive approach; he points out that it hides the history of a concept in order to present a facade of the infallibility of mathematics. But mathematics is not infallible, it changes over time: think of the fact that even the axiomatization of Euclidean geometry did not fully occur until the 19th century (and in fact can we ever be sure that it has so fully occurred?). We would argue that this can be translated over to the deductivist (logical) approach to the judicial decision - it hides the story of just why the decision was arrived at and presents us with a fully-fledged logical decision.

Now, I do not wish to argue that logic is without value simply because there are philosophical problems associated with it. For example, we can use concepts like logic and implication in the mathematical sphere with elegance and power (think of the computational power of computer logic circuits) but that does not require us to believe that they have any life outside of our use of them; they compel us to do nothing - to do that they should at least have the requirement that their interpretation be precise and without debate.

We can conclude, then, that the claim of the judiciary to be bound by logic is frequently made; but our counter-claim is that there exists no logic with the power to bind them. Logic is too relative a phenomenon for that.

## CHAPTER 5

### A SHORT ANALYSIS OF CURRENT LEGAL CONSULTATIVE SYSTEMS

#### 5.1 INTRODUCTION

There have been a variety of computer legal advisory systems produced in the last decade or so, which have attempted to confront the goal set by Mehl to provide a computer system which can answer questions set to it over any area of the law. It is not unkind to the researchers involved to say that no system has so far shown itself to be capable of this extensibility; we do not take the view that the ELI system is presently capable of this - rather that it attempts to solve at least some of the problems which earlier research has brought to light.

The view we take is that any machine which will, eventually, be able to handle legislation in a sound and complete manner - which will not suffer from uneconomic setting up, high maintenance costs etc. - will be an engineered system. We mean by an engineered system that it will not operate with any one particular formalism, but that it will use many of the techniques from computer science which are currently being developed. These techniques may well be software engineering, logic programming, associative networks etc. They will, however, be used in association with each other, combined to form the most efficient and elegant solution to the problem that the period's state of the art can produce.

Since we wish to use these systems to highlight aspects of the ELI system, we shall criticize particular elements of these other systems as required by this procedure, rather than examine all possible aspects of the systems. We are not concerned here so much with the quality or type of advice; rather with the methods that each system uses to represent the law. We argue that this representation is the most important aspect of the systems, at least at this stage in the development of the field. The systems which we shall look at are:

- a) The TAXMAN Project
- b) The MELDMAN Project
- c) The LEGOL project
- d) The INVERCLYDE project
- e) The PROLOG project

There are, of course, several other systems which provide interaction with the legal community (for example [Sprowl, 1980]), but handle "law" less than they provide a useful clerical service - Sprowl's system, for example, uses templates to produce documents but not legal advice; also since Hellawell's system [Hellawell, 1980] uses, broadly, the same data processing techniques as does the Inverclyde project we shall omit it from the discussion here - we examine such techniques when discussing the Inverclyde project.

There is also the approach taken by Foundos with the PRETACS/MERITAL project [Foundos, 1971] with a cybernetic game playing approach:

"Taxpayers and Revenue have been guessing a lot for a long time."

We disagree with (and are confused by) many of the statements of this project; for example:

"Both men and MERITAL speak the same language as defined by the accountant's formula, semantics and information theory."

and also the validity of applying control techniques which have arisen from electrical engineering to social situations. However, a full critique is out of place here.

We deal with the nominated projects in the most convenient order, not in order of importance or success.

## 5.2 THE TAXMAN PROJECT

McCarty's TAXMAN project is interesting both because it is perhaps the longest running project and it is one of the earliest to use "knowledge representation" techniques which have come from the Artificial Intelligence community.

The basic technique which McCarty uses is the associative network or "semantic" network as it is frequently called because of its claimed ability to handle semantic representations.

McCarty has stated that only certain areas of legislation are initially feasible for representation:

"Since the representation problems for a legal consultation system are so difficult, it is tempting to start with the 'simplest' possible legal issues, such as the subject matter of the first-year law school courses. We might therefore be tempted to investigate assault and battery cases from the first-year torts course (see Meldman, 1975), or offer and acceptance cases from the first-year contracts course. But these cases are 'simple' for law students primarily because they draw upon ordinary human experience, and this is precisely what makes them so difficult for an artificial intelligence system. To understand tort cases, we must understand all the ways in which human beings can be injured, intentionally and unintentionally, mentally and physically, with and without justification. To understand contract cases, we must understand



the expectations of real people in concrete business situations, and the ambiguities of human language in expressing particular contractual intentions. If we abstract away these details, we will miss entirely the central features of legal reasoning, and our consultation system will tend to produce only the more trivial results.

Paradoxically, the cases that are most tractable for an artificial intelligence system are those cases, usually involving commercial and corporate matters, which a lawyer finds most complex."

[McCarty, p4, 1980b]

McCarty explains why corporate areas are so tractable by recourse to the invention of legal entities and actions which describe such complex commercial and corporate domains. These actions then become more malleable or "discussable" than human entities or actions. For example, "corporation" is more easily described than, say, "damage" or "intent to injure". McCarty therefore uses these entities and actions within his system, holding them in a "semantic" network.

McCarty's work has been described as "analogical" by Morrisse because:

"Semantic network comparison, like analogical reasoning, involves the comparison of fact situations. In semantic network comparison the fact situations are represented by semantic networks that consist of two types of elements, things and relations. ... In semantic network comparison the computer compares the semantic network of the case with which the attorney is presently concerned with the semantic network of a prior case or statute to determine if the two networks match."

[Morrisse, 1980]

But this is problematical because it assumes that there must be a prior recognition that there is a possibility that the situation under view might accord with a specific model held in the computer system. This also requires that the case to be matched must be represented in a format which is near identical to that held by the system - using similar atomic entities and relationships.

McCarty has pushed this network model to its limits, conscious that such testing illuminates the weaknesses and strengths of the model:

" ... the simplification inherent in a formal model is also the source of its power and utility: it will often lead us to insights that would otherwise be obscured, overwhelmed by the complexity of our data. The unexpected consequences of our formulations may reveal surprising truths or, just as often, the inadequacy of the formulations themselves."

[McCarty, p841, 1977]

McCarty's analysis of the New Jersey/Delaware example does throw light on some problems of the network formalism:

"Some of the facts could not be represented at all in the current TAXMAN system. For example it is not possible to express the fact that the New Jersey company 'had been engaged for many years in the business of manufacturing and selling explosives,' or that the plan of reorganization was 'carried into effect with the assent of a sufficient proportion of the stockholders.' "

[McCarty, p878, 1977]

and:

"One rigidity of the current system is the requirement that a case be described in full, and then expanded in full, before any

analysis mechanisms can be applied."

[McCarty, p885, 1977]

Although semantic networks have been proposed as effective representational systems (for example, [Quillian, 1968]), there have also been criticisms of their claimed power by researchers who are, nonetheless, proponents of their possible utility, for example [Brachman, 1983] and also Woods:

" ... I hope that I have made the point that when one does extract a clear understanding of the semantics of the notation, most of the existing semantic network notations are found wanting in some major respects -- notably the representation of propositions without commitment to asserting their truth and in representing various types of intensional descriptions of objects without commitment to their external existence, their external distinctness, or their completeness in covering all of the objects which are presumed to exist in the world. I have also pointed out the logical inadequacies of almost all current network notations for representing quantified information and some of the disadvantages of some logically adequate techniques."

[Woods, 1975]

The criticisms dovetail neatly with McCarty's perspective on what kinds of law can be handled by AI means. Thus, the criticism that the atomic objects in the formalism must be accepted as "atomic" - the formalism cannot handle "fuzzy" objects or objects which are the construction of social discourse and negotiation.

Secondly, that the links between these objects are not properly "semantic links" - fuzzy relationships cannot be properly (ie "formally" in 'formal semantics') described. For, after all, what is really meant by "is owned by" - surely, this is only understood in terms of the

user's own understanding of the legal context - its usage.

However, McCarty is aware of the limitations of his model:

"I do not believe, however, that this is the final word on the subject. A systematic exploration of the limitations of the current TAXMAN paradigm would, I believe, lead us to a modified paradigm which would correct some of these limitations and permit us to say something about the structure and dynamics of even the more vaguely defined concepts of corporate reorganization law."

[McCarty, p893, 1977]

TAXMAN illustrates well what is perhaps the major problem in attempting to computerize domains which are not fully quantifiable or describable in terms of atomic units. Mehl noted the problem:

".. the language of the law is burdened with synonyms, such as "offer" and "pollicitation", and furthermore these synonyms may cover nuances which are sometimes imprecise and of doubtful use. For example: limitation, expiration, foreclosure, prescription; or again, annul, repeal, rescind. The legal vocabulary sometimes becomes paradoxical. In French, the members of a "societe" are "associés", but the members of an "association" are "sociétaires". It will be seen already that it is essential to give the words a single proper meaning, and eliminate synonyms"

[Mehl, p761, 1958]

Such an attempt was of course advocated by Wittgenstein in his early work; the difficulty (or impossibility) of such an attempt he later acknowledged:

"It is interesting to compare the multiplicity of the tools in language and of the ways they are used, the multiplicity of kinds of word and sentence, with what logicians have said about the

structure of language. (Including the author of the Tractatus Logico-Philosophicus)"

[Wittgenstein, para 23, 1958]

We agree with Wittgenstein and suggest that the logical positivist approach (described in, for example, [Hanfling, 1981]) endorsing the manipulating and reduction of legal language which Mehl proposes is not desirable.

Of course, the success of programs like MYCIN in describing their domain in computer suitable format is impressive when those areas are seen in perspective:

"my friends who are expert about medical records tell me that to attempt to dig out from even the most sophisticated hospital's records the frequency of association between any particular symptom and any particular diagnosis is next to impossible - and when I raise the question of complexes of symptoms, they stop speaking to me. For another thing, doctors keep telling me that diseases change, that this year's flu is different from last years flu, so that symptom-disease records extending far back in time are of very limited usefulness. Moreover, the observation of symptoms is well-supplied with error, and the diagnosis of diseases is even more so; both kinds of error will ordinarily be frozen permanently into symptom-disease statistics. Finally, even if diseases didn't change, doctors would. The usefulness of disease categories is so much a function of available treatments that these categories themselves change as treatments change - a fact hard to incorporate into symptom-disease statistics."

[Edwards, p139, 1972]

The MYCIN system overcomes these epistemological problems in the same way that scientists have in other medical areas (see for example [Fleck, 1979] on the Wasserman Reaction and the socio-genetic origins of scientific "facts") because it has access to quantifiable information; for example, its culture evidence. The importance of this in IKBS research is well realized; Stefik et al [Stefik, 1982] in their 'expert system tutorial' examine the problems which occur when data - including its classification - is not particularly reliable.

Much of what has been called the art of knowledge engineering arises as a direct result of the fact that some computer suitable representation of the domain (ie a normalized model) is necessary before the system has any chance of operating. Feigenbaum's description of the knowledge engineer explicitly describes the process of trying to find a suitable representation:

"She [the knowledge engineer] (in deference to my favorite knowledge engineer) works intensively with an expert to acquire domain-specific knowledge and organize it for use by a program. Simultaneously she is matching the tools of the AI workbench to the task at hand - program organizations, methods of symbolic inference, techniques for the structuring of symbolic information, and the like. If the tool fits, or nearly fits, she uses it. If not necessity mothers AI invention, and a new tool gets created. She builds the early versions of the intelligent agent, guided always by her intent that the program eventually achieve expert levels of performance in the task. She refines or reconceptualizes the system as the increasing amount of acquired knowledge causes the AI tool to "break" or slow down intolerably. She also refines the human interface to the intelligent agent with several aims: to make the system appear "comfortable" to the user ... "

[Feigenbaum, p1017, 1977]

Note though that at no point does Feigenbaum consider here that there might be no AI tool which "nearly fits" or that AI invention cannot provide a suitable representation format.

Buchanan and Headrick wrote of the possibility of the techniques acquired from Heuristic DENDRAL (which we subsume under the heading IKBS) being applicable to the law. We can see the belief - which we do not share - that the "facts" of science accord in format, though perhaps not in complexity with those of law:

"Fact recognition is the first step of the Heuristic DENDRAL program. Given the experimental data, the program, like any scientist must distinguish the "real" data points from the spurious. It separates erroneous data and uninformative pieces of data from the facts that are important for solving the problem. The program can also add pieces of data that "should" have been in the original set based on inferences from theory and the data actually be appearing. ...

Classifying facts and factual situations is also an important part of creative legal research. A concept-formation program written by Hunt and Hovland suggests a means of devising classification rules for a given set of objects. This program models a psychological experiment in learning where an individual is presented with a series of objects described as A's or not A's ... With an extension of the routines used by Hunt and Hovland, one can think of a future computer program that looks at a large set of cases together with their legal conclusions to determine the common elements in the factual situations linking the facts to the conclusions."

[Buchanan, p57, 1970]

We would argue against whether using such an approach in legal systems is possible. The judicial decision (and finding of "facts") seems to us to be slightly more complex. Also, concerning "automatic classification", we should note the scepticism that McDermott encourages in his analysis of just what claims have been made for, and just what results have been arrived at, by AI:

"As a field, artificial intelligence has always been on the border of respectability, and therefore on the border of crackpottery. Many critics ... have argued that we are over the border. We have been defensive towards this charge, drawing ourselves up with dignity when it is made and folding the cloak of Science about us. On the other hand, in private we have been justifiably proud of our willingness to explore weird ideas, because pursuing them is the only way to make progress.

Unfortunately, the necessity for speculation has combined with the culture of the hacker in computer science ... to cripple our self-discipline. In a young field, self-discipline is not necessarily a virtue, but we are not getting any younger. In the past few years, our tolerance of sloppy thinking has led us to repeat many mistakes over and over. If we are to retain any credibility, this should stop."

[McDermott, p143, 1981]

We suggest notions of "automatic classification" are examples of McDermott's use of the term "sloppy thinking".

Perhaps McCarty's use of atomic objects and relationships can be seen as an attempt to come to terms with the fact that law has no quantifiable "cultures" in the MYCIN sense. But law does have legal concepts which are under constant review whenever a new case precedent arises; they are not static in time. Anecdotally, but taken from a newspaper lying



beside me as I write:

"Whatever Mr Montgomery's motive, it was an assault - the law of Scotland has long accepted that kissing a female without her consent is an assault on her and in an age of sexual equality I consider kissing a policeman without his consent must also be regarded as an assault upon him.

Mercifully, the nature, extent and vigor of the kiss planted by Mr Montgomery on Constable Mitchell's lips was not explored in evidence. In the circumstances I shall regard it as an unusual but justifiable act of retaliation which I fervently hope does not become a general practice."

[Sheriff Fulton, written report, Portree Sheriff Court. Quoted in "Press and Journal", Aberdeen, 18th April 1984.]

It is such fuzziness of legal concept ("assault") and continual derivation of new legal instances (kissing "without consent" between males as a new instance of the concept "assault") which makes legal knowledge engineering so very difficult.

### 5.3 THE MELDMAN PROJECT

Jeffrey Meldman [Meldman, 1975] produced a PhD thesis detailing a proposed legal analogical system which was to give advice on cases of assault and battery. The system was never implemented by Meldman, although one other researcher [King, 1976] has done so with one example from Meldman's thesis which includes critical comments.

Meldman's proposed system has been quite heavily criticised by Morrise:

"Unfortunately Meldman's proposed system has severe limitations. Its analysis is based on very simplified doctrines of assault and battery, and its design relies upon its very small knowledge base.

The system also ignores time as a factor and uses artificial categories of its kind hierarchy."

[Morrise p136, 1980]

Is this use of a small knowledge base perhaps the "kludge syndrome" mentioned by Ades above?

What is interesting about Meldman's system, though, is that he proposes that there should be three layers of semantic network to represent the law. The lowest is similar to McCarty's, the other two:

"combine analogical comparison with deductive reasoning. The two higher levels examine the semantic network to see if facts of the case satisfy the general doctrines of assault or battery. For example, if the semantic network contained the elements "plaintive", "defendant", "contact," and intent," and lacked the counter element "consent" the general doctrine of battery would be satisfied."

[Morrise, p135, 1980]

The results of Meldman's work which would be interesting to see would be several. Firstly, how complex is this three tier system; if it operates upon a small number of data items and relationships, will it operate upon a larger number. Secondly, the necessary simplification which occurs when reducing the legal situation to semantic network format is obviously identical to that which occurs with the TAXMAN system - is Meldman's system more simplified because it uses three such levels of representation.

Meldman's view of law seems to be intrinsically hierarchical - the notion that concepts can be of higher or controlling import than others. This view has, in jurisprudential circles, met a substantial amount of opposition. This is though, not to say that such a representation would

not be useful within an engineered legal consultative system.

#### 5.4 THE LEGOL PROJECT

We mentioned earlier that the aims of the LEGOL project transcend the borders of both logic and computing.

Briefly stated, the LEGOL project sets as its targets, the formalisation of logical languages which can express semantic models of the real world. These formalisations are to be interpretable by computer; thus the LEGOL-1, LEGOL-2 (and associated dialects) languages have been devised and analysed in a variety of different situations, each situation being viewed as a domain which exhibits normative aspects - thus:

"LEGOL is a language for writing rules defining information systems at a very general level so that they can be interpreted automatically to discover whether they will have the desired effect. Administrative legislation has the desirable properties of a high-level system specification, providing precise definition of what ought to be done, without saying how"

[Mason, p325, 1980]

Some of the LEGOL research effort has been applied to the study of beauracracic norms [Cook, 1980] and data base philosophy [Stamper, 1977], as well as legislation [Stamper, 1980a]. The wide ranging domain of the LEGOL project arises from the attempt that the research group are making to provide experimental logical formalisms which can model the 'real world' and its normative aspects:

"Information analysis is being studied at the London School of Economics in a study of administrative systems based on complex rules. This, the LEGOL Project, uses statute law as experimental

material. By attempting to devise a formalism which can express the kinds of rules that might appear in a statute defining a tax system, for example, it is hoped to discover a way of specifying an information system at a very general level. A second prototype interpreter for this language is now being designed, an essential part of which is a semantic model. The result of information analysis may be viewed as a semantic model for an application, possibly in some area of law."

[Stamper, p293, 1977]

Stamper sees the semantic model which the LEGOL language is to represent as being a link between the abstract world and the concrete worlds [Stamper, p298, 1977].

The goals, then, of the LEGOL project are the provision of experimental languages which can be used to describe the world in a logical manner. There is also a prescriptive goal - the improving of legal drafting as in [Allen, 1980].

As for the design and provision of 'straightforward' legal consultative, Stamper has described the LEGOL project in relation to McCarty's TAXMAN project:

"The project at the London School of Economics, by contrast, is quite different in emphasis. It is relatively shallow compared with TAXMAN because it deals only with some of the processes of deductive reasoning and it makes no attempt to imitate the essentially human processes of creative argument and judgement. Instead the project is very broad but with a language for doing the analysis."

[Stamper, p52, 1980d]

Even though Stamper sees the LEGOL project having different goals from legal consultative systems which try to represent the judicial process, the LEGOL project has provided research findings which are valuable for this other area. Perhaps the major conclusion which we might take from the LEGOL project is that modelling the law by logical formalisms is by no means as easy as might first strike the computer scientist. Stamper's work supports Allen and Caldwell's statement that "the full scope of the interrelations between logic and law is yet far from clear" [Allen, 1963] by showing many of the problems of handling quite simple legislation (from the point of view of lawyers), but concluding:

"For all its faults, the method of semantic analysis presented here represents a significant advance upon other available techniques. Logical languages which might be used to express rules, such as the predicate language or PROLOG, the computer language derived from the predicate calculus, have not control over the structure of the formulae based upon the semantic theory. The intuition of the analyst is left as the sole guide."

[Stamper, 1980b]

The LEGOL project ran productively until about 1980 when a pause in the research occurred. The project is now restarting, but it is unclear whether exactly the same path will be taken or other logical systems will be appended to newer versions of the LEGOL language. Or so discussion with Stamper has led me to believe might be the case.

## 5.5 THE INVERCLYDE PROJECT

It has been noted by many legal writers that there is a growing quantity of what has become known as "administrative law". For example, Spengler writes:

"Three decades ago already the Lord Chief Justice of England was

warning of 'the new despotism' implicit in the substitution of 'administrative law' for the 'rule of law' and the associated increase in 'the pretensions and encroachments of bureaucracy.' "

[Spengler, p48, 1963]

Spengler's (and the Lord Chief Justice's) thesis is that a large amount of law is being drawn up which exists outside of the common law tradition; this administrative law is coming into being to suit the needs of bureaucracy - clear, well defined procedures with inflexible implementation and little recourse being allowed to common law justice:

"Indeed, so great has become the threat of the administrative juggernaut that in the English-speaking world consideration is being given to adoption of arrangements resembling the office of Ombudsman established in Nordic countries to guard the citizen against the bureaucrat."

[Spengler, p49, 1963]

The existence of this type of administrative law dealing with welfare rights legislation has enabled formalisation of the clear cut (or relatively clear cut) administrative procedures to be handled by a computer system; for example, the Inverclyde project [Adler, 1975] by using common data processing techniques (batch processing and the COBOL programming language) successfully demonstrated that the problem of low take-up of welfare benefits could be solved. This approach, given the new cheaper and more powerful hardware currently available could easily be translated into interactive mode.

We do not feel that the Inverclyde algorithmic approach is suitable for other, more traditional, areas of law. We take this view, not because this approach is so different from others, rather because it does not set itself the goal of modelling more complex legislation.

There is also the interesting problem of updating the monolithic program which is written in a computer programming language. It becomes essential to utilize someone with expertise in both programming and in the area of the legislation which is to be dealt with to properly update the system. This is, of course, a problem which is claimed does not exist with IKBS programs; however, some scepticism seems to be in order here - see Section 9.2 for example. But perhaps this problem is no more severe than that required in amendment of Income Tax programs used by the Inland Revenue because welfare rights legislation has been simplified since the Inverclyde project. Spengler would see this as a further move away from the discretion of common law practice; but systems using this approach need not cope with quite the same numbers of problems as Adler and du Feu faced:

"Probably the chief difficulty in writing the calculation program is the complexity of many benefit assessments. Some of the code is therefore not easy to read and in some cases involving discretion (e.g. how to treat 'essential' HP payments) or unknown factors in the input data (e.g. how many children will be staying at school next year) assumptions have to be made in the program. The program errs on the side of caution when estimating entitlement, and will usually make the most pessimistic assumption. Some other cases are too complex to be tackled by the program (e.g. people with two wives, a student who is head of a household) at least as far as means-tested entitlement is concerned, and where possible in such situations the validation program causes a warning to be printed on the output."

[Adler, p12, 1975]

Hammond has pointed to the revision of welfare rights legislation:

"Computerized descriptions of benefit entitlement are now feasible because the discretionary nature of many benefits has been

abolished."

[Hammond, 1983a]

Of course, some discretion still exists, and there remains the problem of interpretation of the legislation; however, for the computer scientist, the position is very much easier, if not for the client who loses out on discretionary payments.

We are aware of other similar microcomputer based approaches currently in operation, and look forward to their published results.

## 5.6 THE PROLOG PROJECT

The PROLOG project is currently underway at Imperial College, London and, to some extent, is similar to the ELI project. It is in the different attitude to the use of logic versus language which perhaps is the most striking difference between the work at Imperial and the ELI program, so we shall mainly concentrate upon that here.

There has been some comparison between the LEGOL project and the PROLOG project by Sergot [Sergot, 1980], but Stamper has noted the since the versions of LEGOL are so different, such comparison is not straightforward:

"However, PROLOG like LEGOL-2.0 appears to have no place for the kind of semantic structuring that is the main characteristic of other forms of Legol."

[Stamper, 1980b]

The basic assumptions behind the PROLOG project is the work of Kowalski and "logic programming" [Kowalski, 1979] which proposes logic as a problem solving methodology. As one example of this logic implementation on computer systems, Kowalski has shown preference for



Prolog and extensions to it. He points to the power of Horn clauses (of which Prolog is an implementation) as a means of investigating the heuristic search, problem-reduction and program execution models of problem-solving and argues that logical inference provides a model which is simple and powerful. [Kowalski, preface, 1979].

As the ELI system uses a production system formalism, so does the Prolog project - if we accept that the Horn clause formalism is very closely related to that system even though it might be interpreted in a differing manner. Textually, the difference is slight; in the production system the consequent is to the right and the antecedents to the left:

conditions => goal

whereas the Horn clause representation is the reflection:

goal <- conditions

Prolog also allows what have been called degenerate rules, that is rules which have only a goal:

goal <-

which [Cory, p2, 1984] wish to interpret as "facts", that is as conclusions to a rule whose existence rests upon no assumptions. This may well be useful within the programming framework of Prolog and of resolution theorem proving (e.g. [Bundy, 1983]) but is confusing within the framework of mainstream computer science and its emphasis upon data being kept distinct from procedure.

Further to this, Prolog also allows predication of its atomic entities within the rules:

<- male(x)

From the following examples we can see that the project team's normalization of the British Nationality Act ([Hammond, 1983a] [Hammond, 1983b]) is also involved with the Department of Health and Social Security in provision of Supplementary Benefit advice; we concentrate here upon the larger teams work, though) is very close to the formalism which we dealt with in Chapter 4. Instead of rules defining the concept of merchantable quality, they define British Citizenship:

x acquires British citizenship on date y by sect. 1.1

if x was born in the UK

and x was born on date y

and y is after commencement

and x has a parent who qualifies under 1.1 on date y

x acquires British citizenship on date y by sect. 1.2

if x was found as a new-born infant abandoned in the UK

and x was found on date y

and y is after commencement

and not [x was born outside the UK

and not [x was born before commencement]

and not [x was not born to a parent who qualifies under 1.1

at time of birth]

These rules are not actually as used within the system but are relatively near to the more formalized versions. They are easily readable. These rules are the input to an expert system "shell", APES [Hammond, 1983c]; the claimed advantage of an expert system shell is - as noted above - that the interpreting and interface functions of the system do not have to be reprogrammed on each occasion that a new domain is to be handled.

Cory et al intend to normalize the entire Act into this format, a process which they describe:

"The formalisation of the British Nationality Act is an axiomatic theory similar, for example, to an axiomatization of Euclidean Geometry. In principle, any logical consequence of the axiomatization can be generated and tested mechanically by means of a computer-based theorem-prover."

[Cory, p8, 1984]

but later admit:

"The formalisation of the British Nationality Act is actually more difficult than we may have suggested. There are many logical complexities which have not been discussed in detail in this paper."

[Cory, p14, 1984]

We would argue that this difficulty which the project team have found is a direct result of the expectation that a piece of non-administrative legislation can be so easily formalized in a truly logical format. We mean by "a truly logical format" the "axiomatization" for which Cory et al are aiming - the attempt to define a closed logical world which accords with first or second order logic. The comparison of the axiomatization of Euclidean geometry with legal normalization is confusing; what is the connection between the two that the group see? Formalisation in mathematics is still a problematical area:

"... systemization calls for more than the ability of a good librarian. For example, it was not until the nineteenth century that Pasch first formulated axioms concerning the concept "between" which had been tacitly assumed but not explicitly stated in Euclid. Moreover, a field has to often to be developed very thoroughly

before it is ripe for a systematic and rigorous organization. The history of the calculus illustrates the point clearly: founded in the seventeenth century, rapidly expanded in the eighteenth, the calculus got acceptable foundations only in the nineteenth century and even today logicians generally have misgivings on the matter or, like Weyl, still think that analysis is built on sand."

[Wang, p242, 1954]

Thus, even with a formalized system, it is still open to change - or 'negotiation' to give a Lakatosian [Lakatos, 1976] flavour (who was - we would argue - contentiously used to support [Kowalski, p242/4, 1979]).

There are also a variety of ways in which the British Nationality Act can be normalized - the simplest method is to divide the APES rules into smaller rules. For instance, the rule quoted first above could become:

```
x acquires British citizenship on date y by sect. 1.1
    if x satisfies personal citizenship requirements on date y
    and x has a parent who satisfies parental citizenship
        requirements on date y

x satisfies personal citizenship requirements on date y
    if x was born in the UK
    and x was born on date y
    and y is after commencement

x has a parent who satisfies parental citizenship on date y
    if parent of x was a British citizen on date y

etc.
```

Each normalization might well interpret the law in a slightly different manner. Is this the type of formalization which they group are considering, or perhaps just one idealized or Platonic formalization. This seems to go against the view of law as being interpreted by the

judiciary and also against Hart's thesis open texture. The open texture of law can be seen as a matter of observation - legislation is open to different interpretation (no matter how much the legal drafting profession try to cover all loopholes) and we can expect a variety of normalizations. Niblett has noted this and proposed it as an advantage - a variety of expert systems each with a different interpretation of the law which will be suitable for different clients. He writes:

"It is a mistake to assume that one expert system is sufficient for one area of law. Some systems, like some lawyers, will be better than others for they will have a more refined knowledge base and superior powers of reasoning. Just as some lawyers are plaintiffs' men whilst others are more at ease representing defendants, so some systems will be tuned to the requirements of plaintiffs and others to defendants. Some tax systems will favour the Revenue whilst others will be more suited to the taxpayer."

[Niblett, p4, 1981]

We view the law as norm directed and stand within the legal positivist tradition but find it hard to accept some of the assumptions of the Prolog project. For example:

"Logic provides a natural base for a computer interpretable formalism to express legal rules: law treats large sets of complex rules that have long seemed suitable for logical analysis, and once the law is expressed in some appropriate subset of predicate logic, that formulation can function as a program which interprets law."

[Sergot, 1982]

Such a statement ignores the long history of logicians attempts to formalize the law, attempts which were examined in Chapter 4. It is presumptuous to believe that such a naive view of the legal order is sufficient requirement for such a complex task. If it were so, then

logicians would have had very much more success in the field than they have had.

Cory et al quote Laymen E. Allen as recognizing that Horn clause form of logic is a simple form in which to express legislation. The situation is slightly more complex though; Allen did not claim that legislation could be expressed in such a format, only that using such techniques drafting would be made more lucid:

"According to Frank, when courts interpret statutes, they cannot avoid engaging in supplemental law making, for two reasons:

'[T]he necessary generality in the wording of many statutes, and ineptness in the drafting of others, frequently compels the courts, as best they can, to fill in the gaps, an activity which no matter how one may label it, is in part legislative.'

Few would dispute this contention. However, Frank's analysis becomes even more interesting if carried further by pointing out one clear distinction between:

1. the necessary generality in the wording of many statutes
2. ineptitude in drafting

Because of the first, the filling of gaps in legislation by courts cannot and should not be entirely eliminated. However, this is not the case with judicial legislation made necessary by ineptitude in drafting. The contention here is that the necessity for judicial legislation should be minimized insofar as that necessity arises from drafting ineptitude. This Article represents an effort to devise techniques to curtail drafting ineptitude and the ambiguities thereby created."

[Allen, p877/8, 1957]

Therefore we cannot agree that Allen states that the Horn clause form of logic expresses legislation. We would argue that he claimed it might help remove drafting errors.

There is also some contradiction in reports of the ease of inputting rules to the system. Cory et al write:

"The formalisation of legislation by means of rules has almost all the characteristics of an expert system. It differs, however, in one important respect. In a classical expert system, before knowledge can be formalized, it has to be elicited from the subconscious [sic] of an expert. This knowledge elicitation problem is generally regarded as the main bottleneck in the construction of expert systems. It is entirely absent, however, in the case of legislation which is already formulated and written down. Thus the use of expert system techniques for representing legislation has virtually all the advantages of expert systems without the attendant disadvantage of the knowledge elicitation problem."

[Cory, p12, 1984]

But the reported comments of a member of the team seem to belie this:

'Cory explained that "you have to begin with the conclusion, the meaning of the clause; this isn't easy to find or express." .... The student who produced Horn clauses such as these had the most tedious job of all. "The systems analysis is all done on paper- there's no getting away from writing out all those And/Or clauses," said Cory.'

Quoted from Datalink, 13th February 1984

We would argue that the normalization of legislation is not as simple as Cory et al suggest. Rather that it is a very mentally demanding task which requires considerable skill (which legal training develops and encourages). This skill is that which allows the lawyer to interpret legislation (noted above in the extract from Twining and Miers) - but until this has occurred the legislation has no meaning and it might as well be buried in the "subconscious", wherever that might lie.

The importance of these criticisms of the Prolog project are substantial; if we are to build useful engineered systems it is necessary to have clear perspectives of the very real problems which have to be met. If we do not, then we risk being incapable of extending our systems into other legal domains, and such generality must be the aim of legal knowledge engineering. The failure to attend to the problems to be solved is a sign of a poor engineering approach.



## CHAPTER 6

### THE ELI PROJECT

#### 6.1 INTRODUCTION

The project shares one major common assumption with the Prolog project; that is, that legal concepts can be defined by rules.

The claim of the ELI program is, however, that it is based upon an analysis of some of the problems of the domain to be modelled, and that this analysis has provided a specification of what the ELI should, at least in the early versions, be capable of providing. Such an analysis and specification is, we would argue, necessary at the start of any sound legal knowledge engineering project. Of course, the specification might prove less than useful; but then, at least, we have a reasonable conceptual and theoretical framework within which to discuss the failure of the system.

Analysis of the domain has led us to believe that any attempt to describe "the law" with any generality within the confines of a first or second order logical system has, to date, been unsuccessful. We are convinced that there is what might be described as a language of discourse whose domain is the law; that there exists a manner of discussing the law at the "meta-level". It is this language with its descriptions of legal concepts, rules and world which it is important for any legal consultative system to encapsulate, and not Mehl's simplified language of discourse where synonyms are reduced to single

tokens.

A major research goal of the ELI project has been the requirement that the legal expert should be able to directly interact with the system - that is, we wish to degrade the role of the "knowledge engineer" from that of constant attendee to the needs of the expert. We feel our goal should be the provision of sufficiently generalized legal expert systems which allow the expert to formulate, structure and then input his or her "expertise".

Friedland has expressed the desirability of this approach:

"The reasons for providing knowledge entry by the domain expert are as follows. First, both accuracy and completeness of the knowledge base suffer when domain knowledge passes through the filter of a computer scientist, who is not an expert in the chosen domain. Much of the knowledge is complex and subtle, and its purpose may not be immediately apparent to the non-expert. Second, a large knowledge base may be built more quickly when non-experts do not have to be intimately involved in describing each object and rule in the knowledge base. Finally, in building a knowledge base for a program to be used by other experts in the field, an element of trust, as embodied in the name of a known authority, is essential. Professor Kede's description of cloning strategies is more likely to be used and trusted by other molecular biologists than is Peter Friedland's translation of Professor Kede's strategies."

[Friedland, p857, 1981]

In this chapter we describe aspects of the current ELI program, and in later chapter describe how it might be extended, and describe its failings.

While the system has not actually been used in real situations, one legally qualified researcher has interacted with it: her comments are contained within Appendix C.

## 6.2 RULES WITHIN THE ELI PROGRAM

There are several important aspects concerning the rules within the ELI system. Firstly, is the question of just what is contained within the rules; secondly, how can rules be used to define concepts and rights. There is also the question of how are these to be interpreted.

### 6.2.1 Rule Content

We take a similar view to Sergot:

"Finally, the norms expressed in law change so rapidly, and are derived from so many separate sources, that only an experienced lawyer could attempt the construction of a useful legal expert system. To expect him to learn unfamiliar computer languages in addition is unreasonable."

[Sergot, p42, 1982]

Ideally, the language in which the lawyer should be able to express his rules should be as close to the legal language that we suggest exists. The nature of this language is that it is a subset of natural language (but not a "simplification" of natural language); we propose that by using English text within a production system formalism that this legal expression can be easily handled.

The ELI system, as we have already described, allows rules of the following format:

```
IF (applicant is qualified as claimant)
   (applicant is aged 60 or more)
THEN
   (applicant need not sign on as available for work)
```

This rule says:

if the applicant satisfies the rules of entitlement to Supplementary Benefit and is entitled to claim this Benefit and the applicant is over 60 years old, then he/she need not attend the offices of the DHSS to sign the attendance slip in their claim booklet.

There is no specific format for these rule elements; it is only coincidental that many begin with the word "applicant" - this does not operate like a variable within a logical system and is not operated upon by the program.

Other rules can be much longer, the length of the textual string contained within the parenthesis being limited only by the expressive needs of the person who inputs the rules. For example:

```
IF (applicant can claim standard housing benefit)
  (applicant lets part of home to a sub-tenant,
   or to a tenant if a home occupier)
THEN
  (applicant's standard housing benefit rebate/allowance
   will be calculated on rent received from tenant excluding:
    rates
    water rates or charges for 'environmental services'
    payment for meals
    payment for heating, hot water, lighting and cooking
    $2.70 is deducted if furniture is provided
    $1.35 is deducted if services but no furniture is provided
    $0.35 is deducted if applicant lets a garage or outbuilding)
```

There are two points which should be made about this latter rule. It contains quite an amount of information which could easily be calculated by the ELI program (simply by providing a calculation mechanism) but which we have decided not to implement in this version of ELI. Whether it is really useful is a moot point - such simple calculation would not be beyond the power of the advisor who is to use the system. The second point is that it could have been handled by several rules; the "expert" (i.e. the author - his "expertise" in the domain is a controversial question) though decided that it was not essential - that as a rule it

expressed exactly what was required and did not require a formalized format.

We should digress here and discuss what sort of language such systems require.

Feigenbaum has proposed that self advisory IKBS programs will be a major market product within the near future with substantial profits accruing. Under the heading of "New Fields of Application" he writes:

"Home entertainment and advice-giving: AI scientists have long used games as a vehicle for exploring new concepts. A game generally has a constrained knowledge base and is highly structured. There exists not only a considerable amount of competence within the AI community for constructing intelligent game programs, but also a vast consumer potential for home entertainment of this sort. In the author's view, the money-making potential of this market will make microcomputer-plus-TV-based home entertainment the dominant market for expert systems. The concept of home entertainment is broader than games and includes consultation and advice about a broad range of subjects of interest to the consumer (e.g., financial advice, and garden and plant care). Knowledge bases for these specialities will be assembled by experts in much the same way that the 'how-to' books traditionally have been."

[Feigenbaum, p73, 1980]

Whether tort advice will appear on your home TV set is debatable. The Inverclyde project was adjudged successful on technical grounds but not on managerial grounds partly because of the complexities of providing useful legal advice:

"Following the experimental period, the participants agreed that the project could only succeed in the long-term as part of a

comprehensive and outgoing advisory service, but as this was not felt by the local authority to be one of their responsibilities, the experiment was discontinued."

[du Feu, p190, 1980]

It seems that legal advice might of itself not be enough; there must be a contextual interpretation of that advice which takes into account the client's situation. A simple listing of IKBS programs immediately disproves the idea that those presently available are used to interact with non-experts. Rather these systems are used to assist already trained professionals to carry out their roles, albeit in a more efficient or adept manner. The application of IKBS programs in this type of area has perhaps greater effect than in those areas where the system might be used by complete non-experts. This is because those experts are able to assimilate more advice from less knowledge; a pointer to a specific goal may be all that is required to improve greatly the effectiveness of that professional's work. A non-expert moreover, cannot be expected to understand the implications of advice given in relation to the whole domain in which his problem resides without a large amount of explanation; and the more information which the expert system provides, the more it becomes a teach-yourself-system and less like a true consultation system.

The problems of even deciding what is a legal problem are not trivial to the non-expert. The situation, with regard to these "legal problems", has been expressed by Bing and Harvold:

"The problem, as expressed by the client, is not primarily classified as "legal" or otherwise. The first step of the lawyer is to determine whether the problem - as presented to him through his client - is legal or partially legal. This is not as easy as it may appear. If a client complains about health, housing, his family or his economy, it is not evident that the best solution is

an invalidity pension, housing grants, and a divorce. The legal problems may be part of a more complex problem situation - or, indeed, symptoms of other problems. The client might perhaps be better aided by medical care, retraining and contact with a social adviser than by extensive legal assistance."

[Bing, p18, 1977]

Allowing those who have no true appreciation of the entirety of the situation to freely access expert systems, may not be doing them the service to which they are entitled.

A properly engineered IKBS system, we can conclude should design its output for the person who is most likely to benefit from using it.

The ELI system allows the negation of both conditions and goals. These elements can be negated by using English text:

(applicant is not entitled to supplementary benefit)

which is not recognized by the system. Alternatively, the prefix "NIL" (used for convenience in this INTERLISP programmed system) can be used; this is recognized by the system:

(NIL applicant is entitled to supplementary benefit)

which means the same as above. Examples containing double negation are allowable, but confusing:

(NIL applicant is not entitled to supplementary benefit)

The preferred form of negation is with the prefix NIL. Using this allows the system to use the negated information in the processing. The system can also simplify the questions that are being asked by removing the prefix from the rule element before asking the user of their truth. Thus if the element which is to be presented to the user is:

the user will see:

Is this true?

:: today is friday

answering "yes" will cause the system to conclude "no" and answering "no" will cause the system to conclude "yes". Such a facility only helps increase the simplicity of the interface.

We make mention (in Chapter 7) of the use of a token in the knowledge base which handles both the value of the condition or goal and also any annotation which the expert wishes to attach to that token. The usefulness of this is that the expert can attach further information to the rule element either to explain further what the element means or to indicate where it was extracted from. Thus these tokens could be used to reference legislation and/or case precedent. The annotations attached can be of any format - perhaps most usefully in English text - and are available to annotate rules as well as the conditions and goals.

We deal later with the conceptual matching process - a procedure which was designed to aid rule input when the rules are of a textual format.

### 6.2.2 Definitional Rules

We follow the general pattern of the Prolog project (and of that part of the means-end analysis of [Waterman, 1981] which deals with legislation) in that we see that rules can be used to define "legal concepts" which appear in the antecedent list of other rules. Take for example the rule we presented above:

```
IF (applicant is qualified as claimant)
   (applicant is aged 60 or more)
THEN
   (applicant need not sign on as available for work)
```

We can say that this rule means:



if the applicant satisfies the rules of entitlement to Supplementary Benefit and is entitled to claim this Benefit and the applicant is over 60 years old, then he/she need not attend the offices of the DHSS to sign the attendance slip in their claim booklet.

because there are other rules in the system which define just what is meant by "applicant is qualified as claimant". This is not a "legal" concept, but one which was used to represent the fact that before a person can receive Supplementary Benefit, he must be capable of satisfying the regulations, and must also be entitled to actually claim. For instance he cannot usually claim if he is not a British citizen nor a national of an EEC country (although ELI copes with certain other cases); and until recently only the male member of a family was entitled to claim for his wife and family (ELI defines those who cannot apply).

We can thus define those who are "entitled to claim" by showing that the applicant accords with, for example, the following two rules:

```
IF (applicant is available for work)
  (applicant is a British citizen)
THEN
  (applicant satisfies the rules of entitlement to
    supplementary benefit)

IF (applicant satisfies the rules of entitlement to
  supplementary benefit)
  (applicant is over 16 years of age)
  (applicant is usually resident in the UK)
  (NIL applicant is in full-time work)
THEN
  (applicant is qualified as claimant)
```

The same definitional process applies for some of the other conditions in the rules. For example, we could use defining rules to define whether the client is a British citizen or not.

It is simple to add rules which have no legal necessity, but improve the consultation by reducing the amount of information which needs to be input. For example:

```
IF (applicant is aged 60 or over)
```

```
THEN  
  (NIL applicant is under 60 years of age)
```

Although, one must be careful not to cause looping during interpretation of the rules with these type of rules - the problem arises in self-reference where in order to prove a given consequent, other rules which contain that consequent as one of their antecedents exist (during testing these make their appearance as infinite loops). As another example, to include information about retirement is simple:

```
IF (applicant is aged 60 or over)  
  (applicant is female)  
THEN  
  (applicant has reached normal age of retirement)
```

### 6.2.3 Handling Negation

In any useful system which attempts to allow non-trivial rule handling, it is necessary to have interaction between rules other than simple definition. For example, in order to have rules which not only define how to be a member of a certain classification it is necessary to have rules which define who cannot be a member of that classification. This cannot be done by using only negation of the rule's conditions:

```
IF (child is under 5 years old)  
THEN  
  (child is a pre-school child)
```

is completely different from:

```
IF (NIL child is under 5 years old)  
THEN  
  (child is a pre-school child)
```

What is needed is some means of negating the goal. For example:

```
IF (NIL child is under 5 years old)  
THEN  
  (NIL child is a pre-school child)
```

which reads, "if the child is not under 5 years old, then it is not a pre-school child" or "if the child is over 5 years old, it is not a

pre-school child"

The ELI system incorporates this negation handling within the control element. It assumes that a negatively defined conclusion has precedence over a positively defined conclusion (within computing this would be described as "higher priority"). Thus, if the two rules above existed, it would attempt to prove the rule with the negated goal before trying to prove the rule without the negated goal.

Thus with the ELI rule:

```
IF (NIL applicant is disabled)
    (applicants works 30 or more hours per week)
THEN
    (applicant is in part-time work)
```

when it is being interpreted, the ELI interpreter will first try to determine whether the first condition has any rules in the knowledge base which define it (either negatively or positively). It does this by first searching for a goal "NIL applicant is disabled". If one exists it attempts to prove it; if proven then the interpreter tries to remove any rules from the knowledge base which have "applicant is disabled" as their goal - because these rules are now redundant. This procedure is very effective in reducing the number of rules which exist within the knowledge base as possible contenders for future triggering. Thus it prunes the knowledge base, to use computing terminology.

If there had been no incidence of the negated goal "applicant is disabled" in the knowledge base, then the system would try to find one which is not negated, and then try to prove it by testing its conditions. [Clark, 1978] has discussed the use of negation by failure - the searching of a data base for a certain item and if it does not exist assuming that it has been negated. The ELI system has a form of this negation by failure; it comes into play, though, only when a non-negated goal has been found. If all rules which have it as their consequent (ie the "concept's" defining rules) have been unable to prove

it, then the ELI interpret assumes that it has been negated. This negation by failure can be expressed in another way. The knowledge base of the program contains - we might say - all legal rules which might conveniently define given conclusion, say illegal act x. If, given situation y, after processing all these rules, we have not been able to "infer" the conclusion x, then we can colloquially say that, "there's no law against y."

We can usefully talk of the ELI interpreter being in either "open" or "closed" mode. If we assume that the knowledge base contains all possible information and rules defining the domain, then it is closed and negation by failure is a suitable strategy.

If the knowledge base does not contain all possible information, then it is open; and negation by failure is not a suitable strategy because it may well be that there are situations (known to the person who interacts with the system) which are not represented in the knowledge base - ie that there are defining rules external to the knowledge base. In this case, before assuming negation the system should ask the user whether to assume negation by failure.

Switching between the two modes could conceivably be useful (and simply done by amending procedural control); it is not, though, present in the ELI system at the moment. ELI presently uses the closed mode, ensuring that there is a sufficiently general condition which is askable; for instance, the condition:

"circumstances are such that the Supplementary Benefits Officer considers it unreasonable for the applicant to sign on".

#### 6.2.3.1 Pruning The Knowledge Base -

There is a second strategy which utilizes negation. Like the previous strategy it operates within the program, but rather than discussing actual rules we can see it in the following toy example. There are a large collection of productions, of which several have the same conclusion:

<a> <b> => <x>  
<c> <b> => <x>  
<d> <e> => <x>  
<f> <g> <h> => <x>  
...  
...  
etc.

We wish to discover whether <x> can be concluded, and in the traditional form of production system interpreter (perhaps "pure" is a better description) we must try each of these rules in turn until we do trigger one rule or we finally terminate the search and conclude that "we have proven not <x> by failure".

For example, we could list a series of rules which define who is allowed to apply for supplementary benefit. These rules would list the necessary conditions (residence, nationality, age etc.) which allow access to that benefit. If none of the rules applied to the recipient of the systems's advice, we would have had to proceed through all these rules asking questions at each one. And such monotonous question asking is one of the potential faults of advisory systems; the user is bored by its inflexibility.

In real life, though, the situation is quite different. We frequently do not have to resort to such a strategy; we usually have a shortcut method of deciding whether <x> can be effected or not - we have information which can be modelled by a rule of the sort:

$\langle y \rangle \langle z \rangle \Rightarrow \text{not } \langle x \rangle$

An example of this type of rule from the ELI program is:

```
IF (applicant is 19 years of age or more)
  (applicant is in full-time non-advanced education)
THEN
  (NIL claim can be made for supplementary benefit)
```

When we are trying to effect a certain conclusion, we can first try to discover whether we can trigger a rule whose RHS is the negated version of our desired conclusion. If one such "negative" rule is found and triggered then all the other "positive" rules can be pruned from the knowledge base.

Such pruning can quickly reduce a large knowledge base to a relatively small one; this is advantageous because it has been noted that the processing time for production systems is directly related to the number of rules which they contain:

"A production system is a programming language with an unfortunate characteristic: larger production system programs execute more slowly than small ones. The extra instructions in the larger program do not have to execute to slow down the system, their mere presence is sufficient."

[Forgy, 1979]

Such pruning is a necessary tactic, because in legal knowledge based systems we can expect many thousands of rules - the law is after all quite complex - and in such large systems simple negation by failure is a wasteful procedure.

#### 6.2.4 Control

The use of negation in ELI is an example of using rules to control procedural interpretation.

The general situation and necessity for control is succinctly represented by Georgeff:

"A production system consists of a set of modules or procedures called productions and a data base on which these productions operate. Each production denotes a condition-action pair, the conditions having to be satisfied by a state of the data base before the production can be applied to that state, and the action specifying the result of the application. No explicit means of procedural control are provided. The system is thus solely event driven, and production invocation can only be achieved indirectly through the data base.

This lack of control structure results in a system which is strongly modular, flexible and adaptive, and thus well suited as a knowledge-based system. However, in many problem domains some means of representing and using control knowledge is essential. This knowledge can be important in two ways. First, the solution to the problem may require knowledge about plans or sequences of action rather than simply about individual actions... Second, the efficiency of the system and the size of the search space may depend on domain specific constraints on production system invocation."

[Georgeff, p176, 1982]

Georgeff proposes that there should be a separate control structure within the system which determines which productions are to be part of the conflict resolution group at any given time (by conflict resolution group we mean the set of those productions which we could try next - obviously we wish to try the one which is most likely to provide us with

a solution, and also a solution at the least computational expense). For ease of interaction with the system, though, such a separate control module is problematic; it means a more complex structure (which the expert inputting the information) must be aware of.

We prefer to use Davis's approach, where the productions are "turned in on themselves" since this allows the expert to conceptualize in a similar way about the rules and rule handling strategies:

"the control structures and knowledge representations can be made sufficiently comprehensible to the expert (at the conceptual level) that he can (a) understand the system's behaviour in those terms and (b) use them to codify his own knowledge. This insures that the expert understands system performance well enough to know what to correct, and can then express the required knowledge, i.e., he can "think" in those terms. Thus part of the task of establishing the link between the expert and the system involves insulating the expert from the details of implementation, by establishing a discourse at a level high enough that we do not end up effectively having to teach him how to program."

[Davis, p321, 1977a]

We argue that the simple technique of negation within the productions is such a conceptualizable methodology, common in every day and legal discourse. While relatively simple, it allows the user to specify at the meta-level which productions should be removed from the existing list. We do not add any productions to the knowledge base by use of this technique; but if we start with a large enough collection of productions, we can quickly prune them to a set which is suitable for the interaction in hand.



The ELI system operates in a domain where there is no attempt to find any one particular solution; that is, since the ultimate goals being effected are beneficial to the client (in that they represent accruing sums of money) the strategy is to try and effect all those which we can. There is therefore no need to incorporate any quantification of the strength or usefulness of any particular path. We are not certain how any attempt to incorporate such knowledge beside the negation processing of ELI would be achieved, but see it as a possible area for engineering enquiry.

It has been pointed out (for example, [Davis, p219, 1980b]) that elegance is achieved by having the system interpreter interpret both meta and object level productions. Negation, as used in ELI, allows this.

Davis [Davis, p224, 1980b] discusses the distinctions between content reference "by name" versus reference "by description". The former references rules by their name (say RULE-38 in the ELI system); the latter by aspects of the content of the rule. Since MYCIN (used as the basis for TEIRESIAS) used a well formatted rule based upon a well formalized language syntax, it was possible to refer to the rules by the contents of each segment of the condition. Shortliffe [Shortliffe, p113, 1975] represented his language as:

```

<rule> ::= <premise> <action> | <premise> <action> <else>
<premise> ::= ($AND <condition> .. <condition>)
<condition> ::= (<func1> <context> <parameter>) |
                (<func2> <context> <parameter> <value>) |
                (<special-func> <arguments>) |
                ($OR <condition> ... <condition>)
<action> ::= <concpart>
<else> ::= <concpart>
<concpart> ::= <conclusion> | <actfunc> |

```

(DO-ALL <conclusion> .. <conclusion>) |

(DO-ALL <actfunc> .. <actfunc>)

where the terminals are descriptors of the domain in which MYCIN operates.

The main argument of much of this text is that such a representation is not possible within legal areas. Legal knowledge engineering must accept that its problems are, to a very great extent, the problems of natural language handling. Therefore we cannot expect any legal IKBS system to be quite so elegantly designed as Davis' system. However, we can be less specific and can refer to the rules either by one of their conditions or by their goal. In the negation processing, we reference rules "by their consequents".

### 6.3 CONCEPTUAL MATCHING

To overcome the problems inherent in handling textual rules, the tactic of trying to match rule elements by words or phrases contained within them was devised. This process, we believed could be expanded to allow more detailed analysis of the rules in the system. This would, then allow a more advanced form of meta-knowledge by content referencing. However, as we discuss in Section 7.2.4, due to implementation factors (specifically speed) this matching process is very processor intensive; in that section we describe how we might improve it in future versions of ELI.

The current module is relatively simple, but does allow expansion. It operates by using three word lists (an "important", a "noise" and a "permanent noise" list) together with two lists which contain, respectively, conceptual matches and non-conceptual matches. A conceptual match is a match between two rule elements which have a similar meaning but a different textual format; for instance:

(applicant has retired)

and

(applicant is over retirement age)

or:

(a claim can be made for supplementary benefit)

and

(applicant is entitled to supplementary benefit)

and

(rules of entitlement to supplementary benefit are satisfied)

We might say that the first two are similar because of their use of the term "retirement", and the second by the term "supplementary benefit". However, the system might contain other rules which are not conceptual matches which use the same terms; for example:

(retirement age has not been reached)

or

(supplementary benefit is known as supplementary allowance)

We can conclude that there are problems about deciding just which rule elements are matched.

We believe that this conceptual matching is important in any system which will have more than two or three hundred rules. Because there is no set format for the rules and because we wish the program to be able to accept legal language, it is essential that some means of flexibly accepting input is incorporated within the system.

As the rules are input to the system they are examined to ascertain the existence of "important" words (which can be defined in the SETUP mode or added during the BUILD process - see Appendix A) the system will ask the expert whether the input rule element is a match with an already input element. If so, they are treated as equivalent; if not they are distinguished. This information is then kept by the system and whenever the knowledge base is to be updated, this information is still available.

In inputting the rules to the ELI system, the matching module did notice many matches where there were spelling mistakes, differing words etc. From that point of view, the module was effective; it does however work on a simple algorithm and could be much improved.

One early design idea had been to use techniques from the language translation field for example by utilizing a dictionary. One major disadvantage of this methodology is that a good dictionary which is context sensitive requires a substantial input of expertise (for example, [Macklovitch, 1984] states that only 450 terms can be properly indexed per man/year). This input is contrary to the effort that we wish from the eventual users of legal advisory systems.

We intend to continue examining how the ELI concept matching module can be improved; not least, because we believe it could be very useful in any meta-knowledge using reference by content.

## CHAPTER 7

### ELI IMPLEMENTATION

#### 7.1 INTRODUCTION

There are basically three problems in the implementation of an IKBS system. The first, and most difficult, is to choose the data structures which will represent the area of expertise handled by the system. As with all computer programs it is necessary to choose one of several possible structures. Each of the possible data structures might be advantageous in solving one particular problem, but not another. So it is with IKBS programs - one might use production rules, another frames and a third, associative networks. Just as likely as a system which uses only one data structure to handle its expertise is a system which uses more than one (MYCIN uses a tree structure - called a context tree - as its prime data structure; it is built and transformed by production rules held in a different structure).

The second area is that of interpretation - how can the information which is held in these data structures be extracted from it.

The third area is that of filling up the data structures with the information which, when interpreted, is to become the system's expertise.

In this chapter INTERLISP functions are given in Pseudo-LISP format to improve clarity. The actual program listing is somewhat more complex, with necessary exceptional case handling and interface incorporated.

### 7.1.1 Rules And Knowledge Bases

The rules which represent or hold the legal information are of a simple format; they are composed of one or more conditions or antecedents and one goal or consequent. It is useful to consider the antecedents being preceded by the keyword "IF", and the consequent being preceded by the keyword "THEN". The format of each rule is then:

```
IF ((antecedent ) ... (antecedent )) THEN (consequent)
```

These antecedents or conditions represent states which must all be in effect for the consequent to be triggered. More loosely, the conditions must be "true" for the consequent also to be proven to be "true". If, when the conditions in a rule are tested, any of the conditions is found not to be true then that rule's consequent cannot be proven. This does not mean that the consequent is "false" - only that it cannot be effected by that rule.

The conditions and consequents in the rule can have any relevant content; they could simply be symbols or, as in the case of ELI, they can be textual strings which represent legal situations. For example one rule might be:

```
IF ((client's resources are insufficient to meet his
      requirements)
    (client is entitled to apply for supplementary benefit))
THEN
    (client is entitled to supplementary benefit)
```

In some systems these rules incorporate Bayesian degrees of belief and/or a variety of logical operators. In the described system such is not the case; the rules are simple textual strings of the above format.

Whatever the choice of condition format it is important because this format will determine the method of interpretation and the amount of expertise which can be extricated from the corpus of rules. In the ELI system, the textual format was decided upon because of the nature of the knowledge to be represented; most other domains can more usefully use more complex schemas, even though this means more effort in setting up and 'tuning' the system. Davis et al [Davis, 1977b] noted that although the format of MYCIN's stylized rules was essentially "non restrictive" within the domain, there was a tendency towards some rules with awkwardly long and complicated premises, and the creation of desired triggering paths was a sometimes non-trivial task. The rule is the basic unit of the type of knowledge based program using production system techniques; but an IKBS system with only one rule is not particularly useful. A useful system is one which contains about 100 to 200 or more; less than this and the system tends to produce trivial advice. The section of the system which holds this corpus of rules is called the knowledge base.

Combined together into a knowledge base, the rules must be interpreted to produce advice. It is necessary to prove more than one consequent because conditions in one rule are often goals from other rules. In order to effect these former rules, the latter ones must be effected first. The process of interpretation (sometimes called the control cycle) is therefore the attempt to trigger the rules contained within the IKBS program. Just as research effort is being applied to the design of rule formats, so is effort being applied to interpreter design.

There is one more aspect of a rule-based program which is necessary for their operation - a small data base which contains a list of all those conditions and goals which are true (having been input by the user) or proven (i.e. effected by the interpretation process). A second list can contain details of conditions which the user has specifically stated

as being false. This data base is empty at the beginning of the interpretation; as the interpreter tries to effect goals, though, the data base gradually fills up with names of proven and true conditions and goals. The data base is available to the interpreter so that it "knows" what has already been proven. This data base is akin to a psychologist's "short term memory", and in Artificial Intelligence literature on production systems it is frequently described as such.

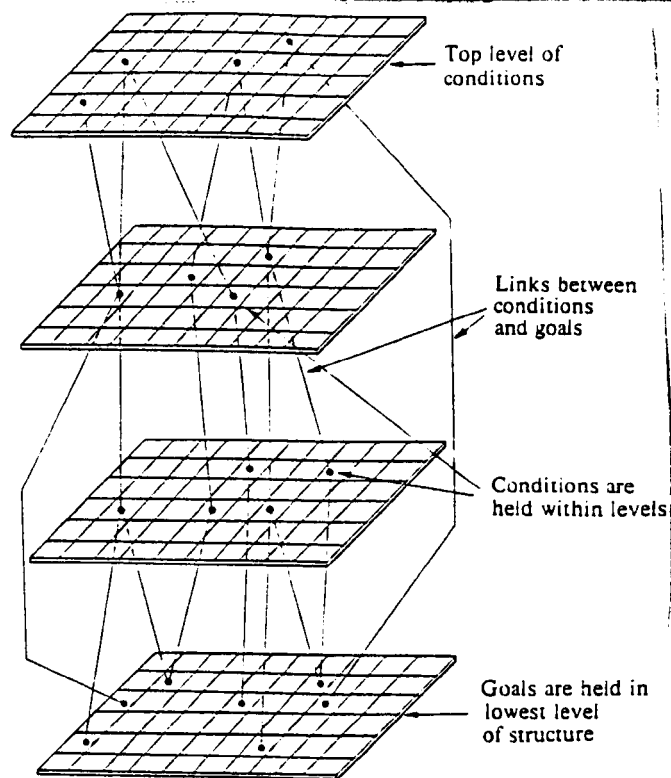
#### 7.1.2 Structuring The Rules In A Graph.

While production systems usually structure their rules into one or more vertical lists, the ELI system utilizes a graph structure with the following aspects:

- (a) the graph has (i) a 'top level' of nodes (ii) 'goal' nodes, and (iii) intervening nodes.
- (b) top level and intervening nodes hold the conditions of rules.
- (c) the goal nodes hold goals of rules.
- (d) nodes are referred to as 'planes' because they frequently hold a condition which is part of several rules.
- (e) planes are ordered into specific levels.

Diagrammatically, the position is:





There are several reasons why this structure was chosen, as an appropriate representation. Elegance was on considered aspect - the fact that the structure can be decomposed into root-topmost or root-bottommost trees (Section 7.2.3.1) but other factors are:

(i) individual conditions can be physically shared by rules

This factor allows conditions to be treated as the most important elementary object. Held in this manner they are examinable in three ways:

- (a) as individual conditions which represent one chunk of causal knowledge
- (b) as parts of an individual rule they represent one cause for that rule having (or not having) triggered
- (c) as common elements of more than one rule, they represent common aspects of the pattern of triggering of those rules.

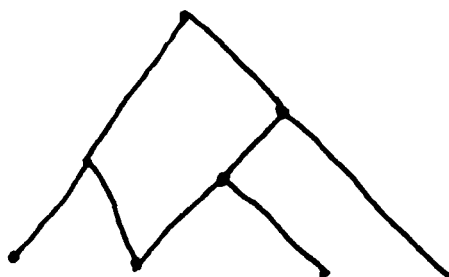
(ii) links allow self control of access

This can be illustrated by considering the topmost condition (or bottommost goal) to define access to the conditions and goals underneath. To a certain extent, the process of structuring productions hierarchically allows the topmost conditions to be defined as meta-conditions, thus in the program one of the conditions for many rules which appears on the top level is "applicant is qualified as client". Such control allows easy pruning of the knowledge base of large blocks of rules.

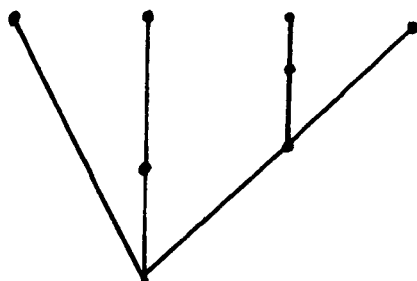
### 7.1.3 Incorporating Rules

Given that a set of rules have been written, how are they incorporated? Somehow each rule must be inserted into the graph structure (or knowledge base as we shall call it from now).

There are three main techniques, which are applied to the rule in the following order; (a) each of the input conditions is matched against the top level conditions until a match between an input and an already assimilated condition is found (if not found, then procedure (b) is followed). The rule number of the input rule is then attached to this condition and the links to lower conditions are retrieved. The same matching technique is then used on the conditions one link below the top level condition. This process continues until no match can be found; at which point the remaining conditions from the input rule are then inserted by themselves. The goal base is then tested to find a match for the input goal; if one is found then the input goal is assimilated with that goal, else the input goal is assimilated by itself. The general pattern of integration from this method is:



(b) If procedure (a) is ineffective (i.e. no match with a top level condition can be found) then an attempt is made to match the input goal with an already assimilated goal. If no match is found then procedure (c) is followed. If a match has been found, however, then the process of (a) is attempted in reverse, i.e. trying to associate the conditions with already assimilated conditions from lower levels upwards. If a point comes when no further matches are found, then the remaining conditions are inserted by themselves, one being placed on the top level. The pattern of integration:



(c) If (b) has not been successful (i.e. no matching goal can be found), then the conditions are inserted as entire rules; i.e. one condition is placed on the top level, the goal is placed in the goal list, and the remaining conditions are placed in the successive levels above the goal. The pattern is:



## 7.2 THE PROGRAMMED SYSTEM

### 7.2.1 Overview Of System Parts

The present system is composed of the following INTERLISP files:

SETUP.COM

BUILD.COM

INTERPRET.COM

KNOWBASE.KNOW

RULES.RULE

(although, in fact, the .KNOW and .RULE files are user created and may have any valid file identifier).

Briefly, they fulfill these roles:

SETUP.COM contains functions which allow the input of rules into RULES.RULE. The rules so input can be edited and suchlike before being automatically inserted into the knowledge base (contained within KNOWBASE.KNOW) by BUILD.COM; this latter file also contains functions for inserting single rules, deleting others and producing a list of conceptual matches between the various rule conditions. INTERPRET.COM allows interpretation (i.e. "consultation") of the knowledge base held in KNOWBASE.KNOW.

## 7.2.2 Implementation Of The Knowledge Base

### 7.2.2.1 Handling Conditions -

The elemental data structure which is used within the system to hold conditions is a list. Before conditions, or more correctly, the planes whose value is a condition, are appended to the structure, the list is of the form

((LEVEL-1) (LEVEL-2) (LEVEL-3) ...)

where the actual number of levels in the initial structure is determined by the user. The rationale for this is discussed later. for now, though, we shall imagine the initial list to be:

((LEVEL-1))

As higher level planes are added to the structure, so must levels (since

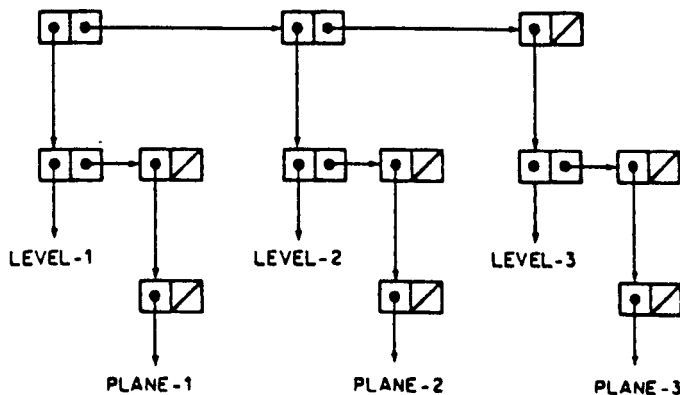
the structure being held on the list is intrinsically hierarchical). Therefore, if a rule with three conditions is added, one will be the value of PLANE-1 (an atom generated by the system); one will be the value of PLANE-2; and the third of PLANE-3. Since each of these planes are particular to one level, there must be three levels within the knowledge base list. i.e. LEVEL-2 and LEVEL-3 (atoms which are system generated) are inserted into the structure.

```
((LEVEL-1)(LEVEL-2)(LEVEL-3))
```

and the planes associated onto the relevant level:

```
((LEVEL-1 (PLANE-1))
 (LEVEL-2 (PLANE-2))
 (LEVEL-3 (PLANE-3)))
```

represented by typical LISP diagram as:



If another rule with three conditions is inserted into the knowledge base, then those conditions are set as the value of newly generated atoms, PLANE-4, PLANE-5, and PLANE-6. They are then attached onto their particular level in LEVELLIST:

```
((LEVEL-1 (PLANE-1)(PLANE-4))
 (LEVEL-2 (PLANE-2)(PLANE-5))
 (LEVEL-3 (PLANE-3)(PLANE-6)))
```

Any planes can be accessed in this structure by processing the desired number of CDR's to access the necessary level; to access the plane once again apply the required number of CDR's.

When entering rules into the knowledge base, if the plane whose value is the only condition of a rule is to be associated with a level, then it is associated with LEVEL-1; if the rule has two conditions then their relevant planes are associated first with LEVEL-1 and then with LEVEL-2.

For example, two rules one with only one condition (the value of PLANE-10, say) and one rule with only two conditions (the value of PLANE-11 and PLANE-12, say) are inserted into the knowledge base. Then LEVELLIST becomes:

```
((LEVEL-1 (PLANE-1)(PLANE-4)(PLANE-10)(PLANE-11)
 (LEVEL-2 (PLANE-2)(PLANE-5)(PLANE-12))
 (LEVEL-3 (PLANE-3)(PLANE-6)))
```

It is important that rules have one condition present on the top level (i.e. LEVEL-1) because the interpreter must be able to access them here.

Conceptually LEVELLIST, in its present state, should be seen as:

|         |         |          |          |
|---------|---------|----------|----------|
| PLANE-1 | PLANE-4 | PLANE-10 | PLANE-11 |
| PLANE-3 | PLANE-6 |          |          |



|         |         |          |
|---------|---------|----------|
| PLANE-2 | PLANE-5 | PLANE-12 |
|---------|---------|----------|

LEVEL-2

Note, that the conceptual ordering of the planes differs from their position within LEVELLIST. The ordering with LEVELLIST is necessitated by the need to add new levels as larger rules (i.e. rules with more than, in the present example, three conditions) are to be inserted. Thus for example, a rule with 4 conditions, where the conditions are the values of PLANE-13, PLANE-14, PLANE-15 and PLANE-16. It is easy to add the first three onto LEVELLIST:

((LEVEL-1 (PLANE-1)(PLANE-4)(PLANE-10)(PLANE-11)(PLANE-13))

(LEVEL-2 (PLANE-2)(PLANE-5)(PLANE-12)(PLANE-14))

(LEVEL-3 (PLANE-3)(PLANE-6)(PLANE-15)))

but what of PLANE-16? The answer is to append a new level (LEVEL- 4) onto the structure, onto which can be attached PLANE-16:

((LEVEL-1 (PLANE-1)(PLANE-4)(PLANE-10)(PLANE-11)(PLANE-13))

(LEVEL-2 (PLANE-2)(PLANE-5)(PLANE-12)(PLANE-14))

(LEVEL-3 (PLANE-3)(PLANE-6)(PLANE-15)

(LEVEL-4 (PLANE-16)))

which diagrammatically can be seen as:

|         |         |          |          |          |         |
|---------|---------|----------|----------|----------|---------|
| PLANE-1 | PLANE-4 | PLANE-10 | PLANE-11 | PLANE-13 | LEVEL-1 |
|---------|---------|----------|----------|----------|---------|

|          |         |
|----------|---------|
| PLANE-16 | LEVEL-4 |
|----------|---------|

|         |         |          |         |
|---------|---------|----------|---------|
| PLANE-3 | PLANE-6 | PLANE-15 | LEVEL-3 |
|---------|---------|----------|---------|

The knowledge base can thus expand to accept rules with any number of conditions, simply by appending them onto the end of LEVELLIST.

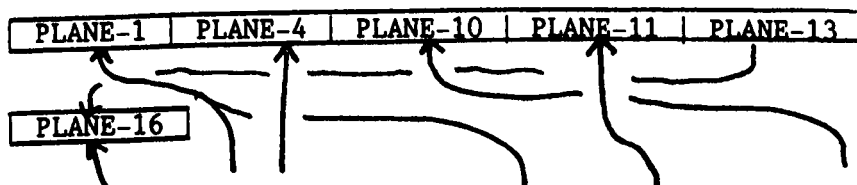
#### 7.2.2.2 Handling Goals -

While it was possible to easily incorporate the goals within LEVELLIST, because goals are treated as conceptually distinct from conditions, the decision was made to incorporate these in a different list structure, GOALLIST. Essentially, though there is no difference between the incorporation procedure: each goal is set as the value of a system generated atom (with the general formal GOAL-n). As each goal atom is generated, it is added onto the end of GOALLIST:

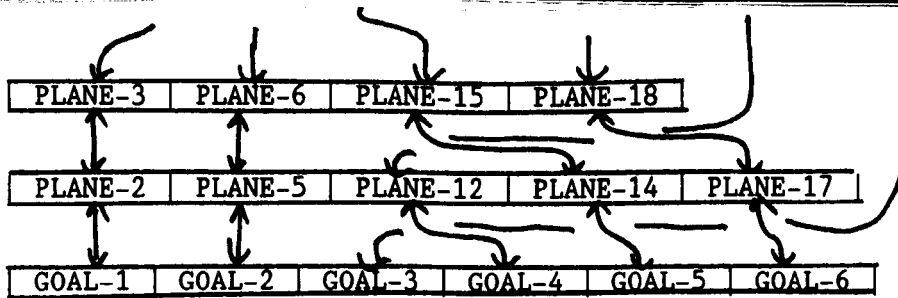
((GOAL-1)(GOAL-2)(GOAL-3) ....)

#### 7.2.2.3 Addressing -

While inserting conditions and goals within these two list structures is relatively easy, there does exist quite a severe problem of keeping tabs on which plane is part of which rule. This problem is exacerbated when two rules have one or more planes in common. Thus if a further rule is to be entered into the structure, and one of its conditions is equivalent to an already assimilated condition on LEVEL-1, then the relevant plane (say, PLANE-1) is utilized as a plane which is part of that 'conceptual rule'. Thus if that rule has conditions on other planes - PLANE-17 and PLANE-18 - we can show the current state of LEVELLIST and GOALLIST, with links connecting elements of each rule as (assuming that each rule has a conceptually distinct goal):







Obviously, the intra-list position of each plane gives no indication of its relevance to planes within other levels. The standard methodology for indicating links between data objects within a data structure is by the one of linked lists or tree structures; but the application of these techniques to the current problem would need a much more complex structure than the format of LEVELLIST; rather than simply indicating the level position, the links would have to point both up and down the structure. This is, of course, possible using Pascal's dynamic pointer ability, if not quite so simple with LISP. However, one major disadvantage with the Pascal implementation is that access to the nodes within the structure is only possible by traversing the graph - the nodes cannot be accessed simply by identifier (which is a LISP advantage I use and explain later).

The solution I applied to the problem utilizes three pieces of information particular to each plane with the structure. These are:

- (i) address of that plane
- (ii) address(es) of ancestor(s)
- (iii) address(es) of descendant(s)

The address of each plane in LEVELLIST is a list composed of two parts - the level, and the position within that level. Thus (1 3) in the example structure we have been dealing with is PLANE-10; (2 2) is PLANE-5 and (4 1) is PLANE-16. An address of (GOAL 1) relates to the first goal on the goal list (i.e. GOAL-1), and (GOAL 5) to GOAL-5 etc.

Each plane holds a list of these addresses for both the descendant(s) and ancestor(s). Thus for PLANE-1 these are:

```
ancestor = NIL
descendants = (3 1) (3 4)
```

For PLANE-3:

```
ancestor = (1 1)
descendant = (2 1)
```

For PLANE-2:

```
ancestor = (3 1)
descendant = (GOAL 1)
```

For GOAL-1:

```
ancestor = (2 1)
descendant = NIL
```

Conditions on the top level and goals have, of course, respectively, no ancestors and no descendants because of their position within the structure.

#### 7.2.2.4 Creating Planes And Goals -

Within INTERLISP, the formation of new data objects is easily achieved, and also the association of a unique identifier with that object. Using Pseudo-LISP code, the function within the described program which achieves this is written:

```
NEWPLANE [ ]
  (SETQ PLANE-VALUE (ADD1 PLANE-VALUE))
  (RETURN (LIST (MKATOM
                 (CONCAT (MKSTRING 'PLANE-)
                         (MKSTRING PLANE-VALUE)))])
```

where PLANE-VALUE is a globally available integer value, incremented on each occasion that a new plane is to be produced.

As mentioned earlier, in the knowledge base, the condition which is to be inserted within the structure is given as the value to the plane; thus the condition (in the form of a list of words) is set as value of the relevant plane. i.e. (SETQ PLANE-1 '(THIS IS A CONDITION))

The other relevant information which is appended to each plane (that is, its address, list of ancestors and descendants, list of rule numbers to which that plane is associated, and interpreter information - dealt with later) are all held on the property list of that atom. We shall examine these in the following sections. Goals are of an identical form to conditions planes, excepting their form is GOAL-n.

#### 7.2.2.5 Structure Traversal -

Traversal of the knowledge base structure is important for both building the knowledge base and also for interpreting it. The process utilizes items held on the property list of the plane known as the LINKDOWN and LINKABOVE properties. Other non-permanent traversal information is also used. This non-permanent information is also held on the property list of atoms with PLANE-n or GOAL-n identifiers. The non-permanent traversal information consists of the following properties lists of addresses:

INTERPRETUP - ancestors of plane or goal

INTERPRETDOWN - descendants of plane or goal

The function RESETKB, used in interpretation, can be set out in pseudo-LISP code as:

```
RESETKB[ ]  
[For all planes and goals in knowledge base  
DO  
  (PUTPROP x 'INTERPRETUP  
    (GETPROP x 'LINKABOVE))  
  (PUTPROP x 'INTERPRETDOWN
```

```
(GETPROP x 'LINKDOWN))]
```

It sets, as initial values, the properties INTERPRETUP and INTERPRETDOWN to be, respectively, identical to LINKABOVE and LINKDOWN. Thus when system interpretation starts, the interpreter (which accesses INTERPRETUP and INTERPRETDOWN but not LINKABOVE and LINKDOWN) has full information allowing it to traverse the entire graph. To reduce the search space within which interpretation is constrained, information (i.e. particular ancestor and descendant addresses) can be removed from these two traversal properties. This removal is done by the function MARK which accepts as its sole argument, a list of addresses for one particular rule (the list is extracted by GRULE which searches the top level planes for the specified rule number and follows the LINKDOWN links returning the addresses of the relevant planes); thus in the previous example this list for rule number 1 (the first inserted) will be:

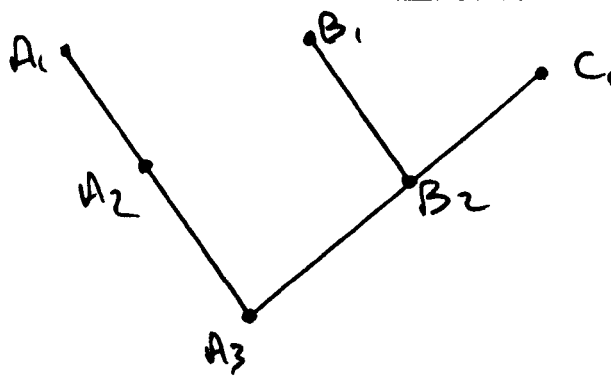
```
((GOAL 1)(2 1)(3 1)(1 1))
```

MARK uses this list to remove from the properties INTERPRETUP and INTERPRETDOWN all ancestor and descendant addresses:

```
MARK [address list for particular rule]
[FOR x in address-list
DO
  (PUTPROP x 'INTERPRETDOWN
    (LDIFFERENCE (GETPROP x 'INTERPRETDOWN)
                  address-list))
  PUTPROP x 'INTERPRETUP
    (LDIFFERENCE (GETPROP x 'INTERPRETUP)
                  address-list))]
```

(where LDIFFERENCE returns the difference between two lists).

Such a marking procedure is essential, because where a goal has been effected it is normally useless to keep on trying to effect it in different ways (that is useless for normal interpretation - not for accessing structural information in a question/answer mode). As an example take the following decomposed tree:



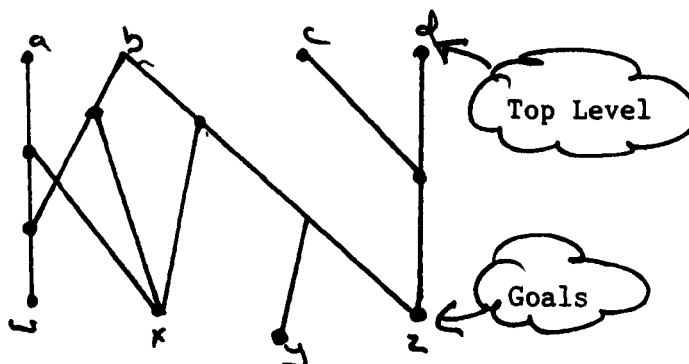
If A1 A2 A3 is followed, then it is no longer essential to test the paths which lead from B1 to B2 to A3 and from C1 to B2 to A3. Therefore MARK is applied to A3 to remove from B1 B2 and C1 all traversal information pertaining to A3. Unless these conditions are intrinsic to other rules, they will not be tested by the interpretation process until after the system is initialized by application of RESETKB.

### 7.2.3 Interpretation

The action of the interpreter can be explained by the use of two procedures, called DOWN and UP and the fact that a complex graph of the sort used in the knowledge base can be decomposed into a variety of tree graphs.

#### 7.2.3.1 Decomposition -

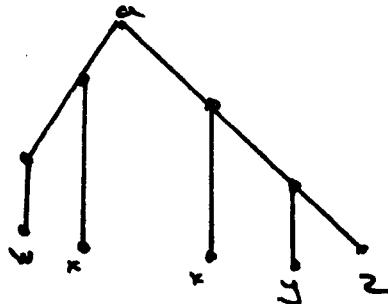
The knowledge base structure can be seen by the following simplified structure:



In order to process this structure, it is decomposed into simple trees. Thus, if a is to be first tested, the search tree below this is:

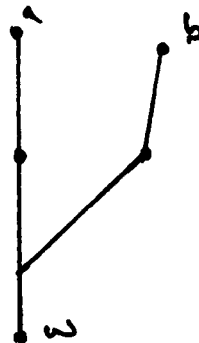


the search tree for the node b to be tested is:

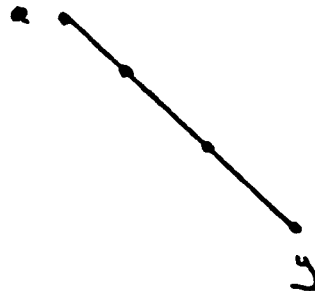


Note that is matter of indifference whether or not x is represented twice as a leaf node with the tree. Likewise, the decomposition will produce trees whose leaf nodes are above the origin. For example from w:

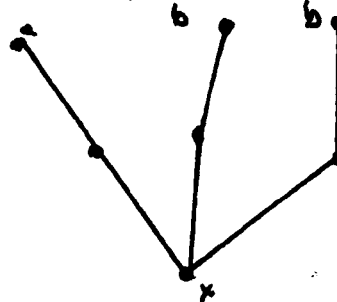
w:



and y:



(which is a tree where every node has only one successor), and x:



Which of these structures are being processed at any time depends upon the point in interpretation that the interpreter procedure is at.

DOWN is the downward processing search procedure. For each of the top level conditions this function is applied to them and their successors. It takes the general form of:

```
DOWN [SUCCESSORLIST]
  IF (CAR SUCCESSORLIST) = leaf node
  THEN
    Add to known goals
    (IGNORE (GETPROP (CAR SUCCESSORLIST) 'LINKABOVE))
    (SETQ SUCCESSORLIST (CDR SUCCESSORLIST))
    (DOWN SUCCESSORLIST)
  ELSE
    IF NOT (TRUE? (CAR SUCCESSORLIST))
    THEN
      (SETQ SUCCESSORLIST (CDR SUCCESSORLIST))
      (DOWN SUCCESSORLIST)
    ELSE
      (DOWN (SUCCESSORLIST of (CAR SUCCESSORLIST)))
      (DOWN (CDR SUCCESSORLIST))
    ENDIF
  ENDIF
```

where SUCCESSORLIST is the list of a node's descendants. The procedure is, of course, a simple recursive process attempting to visit as many leaf-nodes as possible. The search along any particular branch is allowed or disallowed only depending upon the truth or falsity of the condition held on each branch, or whether any of the nodes have been "marked" by the function IGNORE which simply applies MARK to all the rules whose goal has just been proven.

TRUE? is a function which determines the truth or falsity of the particular successor node under consideration. The pseudo-LISP code is:

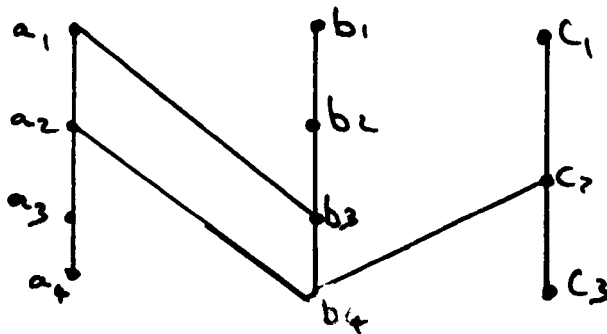
```
TRUE? [CONDITION]
  IF CONDITION already known to be false
  THEN
```

```

                (RETURN NIL)
ENDIF
IF CONDITION already known to be true
THEN
    (RETURN T)
ENDIF
IF CONDITION is also a GOAL
THEN
    IF (UP (ancestors of GOAL)) is true
    THEN
        (RETURN T)
    ENDIF
ELSE
    Ask user and return T or NIL
    as appropriate.
ENDIF.

```

The important point about this function is its use of the function UP when a GOAL is conceptually identical to the CONDITION under consideration. This accords with the use of the rules, as mentioned earlier, in a definitional manner. The possible paths might be:



If a3 is conceptually equivalent to b4, then the path a1 a2 a3 might be taken at which part the interpreter notes that a3 = b4. The function UP is then applied to b4, and since a2 and a1 are known to be true, a3 must also be true (since b4 has been effected; i.e. it has been proved "true".) Therefore the rule a1 a2 a3 a4 is effected, with the goal a4 also being effected.

The pseudo code for UP is almost identical to DOWN, as may be expected since it is only an inverted tree search. However, while DOWN searches for all possible leaves, UP must terminate when the first leaf is achieved. Since it is theoretically impossible to terminate a truly recursive procedure recourse must be made to the LISP stack to allow UP to return T, when a leaf is achieved. The pseudo code is:

```

UP [SUCCESSORLIST]
IF (CAR SUCCESSORLIST) = leaf node

```



```

THEN
    manipulate stack,
    terminate procedure and
    (RETURN T)
ELSE
    IF NOT (TRUE? (CAR SUCCESSORLIST))
    THEN
        (SET SUCCESSORLIST (CDR SUCCESSORLIST))
        (UP SUCCESSORLIST)
    ELSE
        (UP (SUCCESSORLIST of (CAR
                               SUCCESSORLIST)))
        (UP (CDR SUCCESSORLIST))
    ENDIF
ENDIF.

```

SUCCESSORLIST in this case is derived from the LINKABOVE properties rather than - as in DOWN - the LINKDOWN property list.

#### 7.2.3.2 Negation -

In many production systems, negation is not fully utilized. In the ELI program negation is used in several ways. Both conditions and goals can be negated within the system by prefixing them with the flag 'NIL':

(NIL this condition is false)

The ASK user function (called by the TRUE? function) notices this negation and uses it to simplify the questions that are being asked by removing the prefix from the rule element before asking the user of their truth. Thus if the element which is to be presented to the user is:

(NIL today is friday)

the user will see:

Is this true?

:: today is friday

answering "yes" will cause the system to conclude "no" and answering "no" will cause the system to conclude "yes".

The ELI system also incorporates this negation handling within the interpreter. It assumes that a negatively defined goal has precedence over a positively defined goal.

When the rule:

```
IF (NIL applicant is disabled)
  (applicant works 30 or more hours per week)
THEN
  (applicant is in part-time work)
```

is being interpreted, the ELI interpreter will first try to determine whether the first condition has any rules in the knowledge base which define it either negatively or positively - that is, it is negatively defined if the negated version of the condition body appears as a goal; it is positively defined if that condition body appears as a non-negated goal. It does this by first searching for a goal "NIL applicant is disabled". If one exists it attempts to prove it; if proven then the interpreter tries to remove any rules from the knowledge base (with the IGNORE function) which have "applicant is disabled" as their goal - because these rules are now redundant. This procedure is very effective in reducing the number of rules which exist within the knowledge base as possible contenders for future triggering.

If there had been no incidence of the negated goal "applicant is disabled" in the knowledge base, then the system would try to find one which is not negated, and then try to prove it by testing it's conditions. If it is proven, then the negated goal is IGNOREd (that is, rules with it as their goal are removed from the knowledge base).

In order to incorporate this negative processing, the function TRUE? described above must be extended:

```
TRUE? [CONDITION]
IF CONDITION already known to be false
THEN
  (RETURN NIL)
ENDIF
IF CONDITION already known to be true
THEN
  (RETURN T)
```

```

ENDIF
IF CONDITION is negated
THEN
  IF there is a negated corresponding GOAL
  THEN
    IF (UP (ancestors of GOAL)) is true
    THEN
      (RETURN T)
    ENDIF
  ENDIF
ELSE
  IF there is a non-negated corresponding GOAL
  THEN
    IF (UP (ancestors of GOAL)) is true
    THEN
      (RETURN NIL)
    ENDIF
  ENDIF
ELSE      ;CONDITION is not negated
  IF there is a negated corresponding GOAL
  THEN
    IF (UP (ancestors of GOAL))
    THEN
      (RETURN NIL)
    ENDIF
  ENDIF
  IF there is a non-negated corresponding GOAL
  THEN
    IF (UP (ancestors of GOAL))
    THEN
      (RETURN T)
    ENDIF
  ENDIF
  Ask user and return T or NIL as appropriate.

```

#### 7.2.4 Conceptual Matching

It is necessary to apply some kind of conceptual matching to the conditions of input rules as they are input to ensure that the conditions which are conceptually equivalent but not textually equivalent are associated together. The general format of the operation was specified before. There are, within the system, two lists, CONLIST and NOTCONLIST, which respectively hold information concerning conditions and goals which are conceptual matches and secondly concerning conditions and goals which are not conceptual matches. Before rules are entered into the system these two lists are empty. However, as the system discovers that some conditions and some goals can be classified by the relationships "conceptual match" or "not conceptual

match" it adds them to the appropriate list. Having added them to these lists, the same relationships need not be computed in future. These lists are also added to during the "test" phase of knowledge base consultation.

The computation of the relationship is as follows, by CONMATCH whose two arguments are the conditions or goals to be tested for a match:

```
CONMATCH [ST1 ST2]
  IF ST1 ST2 ON CONLIST
    (RETURN T)
  ELSE
    ST1 ST2 ON NOTCONLIST
    (RETURN T)
  ELSE
    IF (CHKLIST ST1 ST2)
      ask user if conceptual match exists
      IF user answers "yes"
        (PUTONCONLIST ST1 ST2)
        (RETURN T)
      ELSE
        (PUTNOTCONLIST ST1 ST2)
        (RETURN NIL)
      ENDIF
    ENDIF
    (RETURN NIL)
  ENDIF
```

PUTONCONLIST and PUTNOTCONLIST are used to keep, respectively, CONLIST and NOTCONLIST up to date. CHKLIST is the heart of the matching process; it is:

```
CHKLIST [X Y]
  Remove noise words from both X and Y
  IF (EQ X Y)
    THEN
      (RETURN T)
    ENDIF
  IF X smaller than Y in length
    THEN
      (SETQ SMALL-LIST X)
      (SETQ LARGE-LIST Y)
    ELSE
      (SETQ SMALL-LIST Y)
      (SETQ LARGE-LIST X)
    ENDIF
  LOOP
    FOR any letter in SMALL-LIST which
      appears in LARGE-LIST
    DO
      IF that letter starts a word or part of a word
        which appears in LARGE-LIST
      THEN
        (RETURN T)
      ENDIF
```

```
REPEAT until no untested letter match between
      SMALL-LIST and LARGE-LIST remains.
(RETURN NIL)
```

The reference to the removal of noise words is to those words like "IS", "T0" etc.

#### 7.2.5 Building The Knowledge Base

When the rules which the user wishes to input into the knowledge base have been entered into the system by the SETUP process, they are input into a PRETTYDEF (i.e. INTERLISP format) file (RULES.RULE) from which they can be read as separate S-expressions.

There are three ways in which rules can be entered into the system, two methods using rules setup in RULES.RULE, and the third allowing extra rules to be input during the build process - particularly useful in "fine-tuning" the knowledge base.

The first input method is to use the rules straight from the file: this is done by FROMFILE:

```
FROMFILE [  ]
  [(LOAD 'RULES.RULE)
   (for x in RULE-COLLECTION do
    (INCORPORATE X))]
```

where rule-collection is the list of rules held in RULES.RULE. INCORPORATE is the "control function handling rule input by means of GUPINC, TDINC and INSPR which add rules, respectively,

- (i) for the goal up
- (ii) for the top down
- (iii) above

as described above. The second method is by using the function READYMADE which can specify any of the existing rules in RULES.RULE and enter them; once again useful in "fine tuning" the knowledge base. The third, uses the same basic finish as is used in SETUP to allow

individual rules to be written and inserted into the knowledge base.

As these rules are input the rule number is incremented so that each rule has a higher number than all previously input rules. This does lead to some lack of neatness - especially when some rules have been input and then deleted; this can be overcome by using the Reformat function which will clear up the knowledge base and overcome "gaps" in the numbering.

The actual setting up of the knowledge base is the adding of atoms to LEVELLIST, together with a PLANE-n identifier, LEVEL-n identifier (or GOAL-n identifier), rule number and lists of ancestors and descendants using GUPINC, INSPR and TDINC. The skeletal LISP format for TDINC is given; the others have the same general characteristics.

```
TDINC [INPUT-RULE RULE-NUMBER]
  (SETQ CONDLIST (CADR RULE))
  (SETQ TOPLIST (MAPCAR CONDLIST 'TOPSER))
  IF (NULL TOPLIST)
  THEN
    (RETURN NIL)
  ENDIF
  (ASSCOND (CAR TOPLIST))
  (SETQ CONDLIST (REMOVE (CAR TOPLIST) CONDLIST))
  (SETQ INCORPORATED-LIST (LIST (CAR TOPLIST)))
  LOWERLEVEL
  (SETQ CLIST CONDLIST)
  (SETQ POSSIBLE-MATCHES
    (GTLKSTO (LAST INCORPORATED-LIST)))
  (SETQ LOWER-CONDS (MAPCAR POSSIBLE-MATCHES 'GTFADD))
  (For Y in LOWER-CONDS
  DO
    IF (CAR Y) = 'GOAL
    THEN
      (SETQ LOWER-CONDS (REMOVE Y LOWER-CONDS))
    ENDIF
  IF (NULL LOWER-CONDS)
  THEN
    (GO ANYMORECONDS?)
  ENDIF
  NEXTPOSSIBLE
  (For ZZ in CLIST
  DO
    (For ZZZ in LOWER-CONDS
    DO
      IF (CONMATCH ZZ ZZZ)
      THEN
        (ASSCOND ZZZ RULE-NUMBER)
        (SETQ INCORPORATED-LIST
          (APPEND INCORPORATED-LIST (LIST ZZZ)))
        (SETQ CONDLIST (REMOVE ZZZ CONDLIST))
      ENDIF
```

```

        until condition associated)
    until condition associated)
    IF condition associated
    THEN
        (GO LOWERLEVEL)
    ENDIF
ANYMORECONDS?
    (For X in CONDLIST
    DO
        (SETQ LEVEL (ADD1 LEVEL))
        (INSCOND X LEVEL)
        (SETQ INCORPORATED-LIST
        (APPEND INCORPORATED-LIST (LIST X))))
    INSERTGOAL
        (SETQ GOAL (CAR (LAST INPUT-RULE)))
        IF GOAL already on GOALLIST
        THEN
            (ASSGOAL GOAL RULE-NUMBER)
        ELSE
            (INPUTGOAL GOAL RULE-NUMBER)
        ENDIF
    Set links between inserted and associated conditions
    and goals using INCORPORATED-LIST
    (RETURN T)

```

TOPSER searches the top level condition to determine whether its argument has a conceptual match there. GTLKSTO simply gets the LINKDOWN property list from a plane. GTFADD uses the address which is its argument to find the name of a plane. ASSCOND is used to associate an input condition with an already existing plane. INSCOND associates the input condition on a new plane which is entered into the knowledge base (i.e. LEVELLIST).

#### 7.2.6 Annotating The Knowledge Base

Annotations can be applied to the knowledge base. These take the form of pieces of text in the form of lists; the lists are formed by the function INPUTTEXT, using both upper and lower case, with paragraphing if desired. The function PPTTEXT is used whenever these annotations are to be printed on the user's terminal.

There are three types of annotation:

- (i) attached to an entire rule
- (ii) attached to one condition

(iii) attached to one goal.

Currently, the annotation is attached to a rule by means of INTERLISP Hash-links; this allows a given atom to be related to another atom in an array. In the described system, the value of the second atom is the annotation. This method was utilized because, in the knowledge base, there is no entity which relates only to a rule as there is to conditions and goals (i.e. the latter are planes with the structure).

The annotations to conditions and goals are held on the property list of those planes with those conditions and goals as values. This is currently suitable since annotations on the conditions are not particularly long or complex. However, this is an area of possible future development, especially if the system is to be used as a commentary upon legislation other than welfare rights.

#### 7.2.7 System Tools

While it is feasible that all rules to be input into the system are initially prepared and input without any recourse to amendment - be it deletion or alteration of the rules - it is more than probable (if not definitely the case) that various alterations will have to be made to ensure correctly derived triggering paths.

One tool available to help search through the knowledge base is the SEARCH procedure which allows:

- (i) one or more incidents of a condition which contain a phrase  
or word to be found
- (ii) one or more rules to be deleted which contain a specified  
condition to be found.

It has the form:

```
SEARCH [string to be searched for]
  (for X in Rules in RULE.LSP
  DO
```



```

      (SETQ TEMP X)
      (SETQ FLAG
        (STRPOS Y (EVAL X)))
    repeatuntil (NOT (NULL FLAG))
  finally
    IF FLAG = T
      (PRIN1 Z)
    ELSE
      (PRIN1 "string not found")
    ENDIF

```

Various editing procedures are also contained within the system. They continue the INTERLISP notion of trying to integrate tools within the environment. Therefore rule, condition and annotation editing all access the INTERLISP editor.

### 7.3 A NOTE ON LANGUAGE

No claim can currently be made to the system being efficient, due in a large part to the fact that it was programmed in INTERLISP. For example, the use of EQUAL rather than EQ is a case in point. Both these functions carry out roughly the same task - testing if two S-expressions are equal. However, since EQ enquires whether the pointers point to the same objects and it is impossible to dump the knowledge base to a file and reload it with the same pointers pointing to the same objects, the program must utilize EQUAL which is substantially slower.

String handling in INTERLISP is also particularly slow - and since so much of the ELI processing is string handling, the slowness of the system can be understood.

While other expert systems have also been produced in this type of high level language (but those use pattern matching rather than string processing techniques) it is by no means necessary - Pascal or Fortran could be used just as well. It is perhaps interesting, though, to briefly note just why we implemented the programs in INTERLISP rather than, say, Pascal.

The major difference between INTERLISP and more procedural languages is the level of abstraction from the machine - LISP objects, collected into lists, are the prime data entities. These objects can be built and manipulated in much more complex ways than what have become known as 'middle level' languages - Algol like languages and their perspective of data as numeric, character and pointer. The power of LISP can be seen from the ability of data to be both data and executing sub program. As well as this complexity of data structure, the interactive nature of INTERLISP allows easy programming and debugging and the ability to break into a program's execution, edit it and then continue execution with the edited program; also available are system packages which can be utilized by programs - the editing functions in the ELI program use the INTERLISP structure editor for instance.

Given these advantages, though, the system is inefficient in comparison with middle level languages. The ELI system was written as a prototype and experimental vehicle - efficiency was not a consideration whereas ease of "bread-boarding" ideas was. I would therefore see that any future development of ELI would require translation into a middle level language now that techniques and methods have been tested.

Some arguments have been proposed for the use of logic programming as the de rigueur language for IKBS research. Kowalski, for one, has put his position as:

"There is only one language suitable for representing information - whether declarative or procedural - and that is first-order predicate logic. There is only one intelligent way to process information - and that is by applying deductive inference methods.

The AI community might have realized this sooner if it weren't so insular. The data base community, for example, learned its lesson several years earlier."

[Kowalski, p44, 1980]

We see this as harking back to the Vienna School and logical positivism, the Platonic belief in the certainty of mathematics [Kline, 1980] and we are not convinced that Kowalski's case is proven; we remain with traditional programming techniques.

## CHAPTER 8

### FURTHER RESEARCH AND DEVELOPMENT WITH RULE-BASED TECHNIQUES.

#### 8.1 INTRODUCTION

We shall severely criticize rule-based legal IKBS systems in the next chapter and propose that for 'real life' law they are severely limited. Of course, that does not mean that these rule-based systems can have no utility at all - simply that this utility is not that for which was originally aimed. The role of these systems, it seems to us, to be appropriate in only clerical areas - areas where rules are applied in a formalistic manner e.g. in bureaucratic situations. Any advice which can be got from them is limited to advice on "how to apply the rules". This, we argue in the next chapter, is not law.

However, we should not write off rule-based systems completely; there is still a substantial market for easily accessible information about how a government department, for example, applies the rules. But, if we wish to argue against these rules, we should still go and see a lawyer.

In this chapter we accept the limitations and posit how, even with them, we might develop the rule-based system described earlier in the text.

#### 8.2 SHORT TERM DEVELOPMENT OF THE ELI PROGRAM

One immediate concern is primarily to carry out some program improvements which analysis of the current version of the program and

plans which already exist have shown to be useful developments. These can be discussed under the headings of:

- a) extending the program to cope with the more complex meta-processing required
- b) extending the program by implementing an assertion facility and an explanation facility
- c) upgrading the importance of the SETUP facility - enabling it for instance, to check rules before input to the knowledge base.
- d) examine the feasibility of utilizing an algorithmic language.

#### 8.2.1 Extending Meta-Level Control

In earlier sections, we noted that one of the current areas of research in IKBS is the introduction of means of meta-level control over interpretation of the rules in the knowledge base. This has been described as allowing control by reasoning about the content of the rules. We have also described the problems which ELI has in deciding just what the content of its rule elements are because those rule elements are based in natural language. We have also pointed out, though, that ELI utilizes what might be described as reference by consequent - used especially in the negation handling interpretation process. Here, we wish to describe some further strategies which we intend to incorporate within the ELI program which allow the expert to insert rules within the knowledge base.

We noted above that there are basically two methods of incorporating meta-rules within an IKBS system. Georgeff [Georgeff, 1982] proposed the use of a separate control module, whilst Davis [Davis, 1980b] proposes that elegance is achieved if the meta and object level rules are processed by the same interpreter. This latter is the method which

we use in the following.

We start by adding an extra rule element keyword which can be recognized by the interpreter - the TRY keyword. This keyword can appear only in goal elements:

```
IF (this is a condition)
    (and this)
THEN
    (TRY this goal)
```

The TRY keyword specifies which goal an attempt should be made to effect. Thus, when the interpreter meets a goal which is prefixed by TRY, it will test whether the antecedent conditions are true, and if so, will try to effect goals in the knowledge base which are of the textual format which follow it. The type of strategic information which we might wish to add to the knowledge base might be:

```
IF (NIL applicant is entitled to supplementary benefit)
THEN
    (TRY applicant is entitled to urgent needs payment)
```

which says that:

"even if the applicant is not entitled to supplementary benefit it is worth trying to ascertain whether an urgent needs payment can be got."

Thus, this kind of rule does not actually decide anything - it only allows the expert to guide the search strategy. This is, of course, an important aspect in systems with large knowledge bases which might contain several thousand rules.

Since these TRY rules are handled by the interpreter, their failure will aid the pruning of the rule base, just as their success will aid the move towards the desired goal.

The ELI interpreter presently begins processing by accessing the conditions in the top level of the knowledge base. Adding these TRY rules requires amendment of this - the interpreter should look first at the goals in the knowledge base and attempt to effect the TRY rules, and only after these have been removed (pruned) from the knowledge base should the top level conditions be serially processed.

This TRY strategy might also usefully be applied if the knowledge base was to be partitioned into modules, each module containing a self contained block of rules. In this case, the TRY rules would reference those blocks of rules rather than other consequents.

This idea of adding keywords to the rule elements seems to be particularly powerful, substantiating the view that logic programming languages (e.g. PROLOG) which do not allow the user to manipulate the interpretation procedure are perhaps not the most effective programming languages for IKBS research. As another example of this use of keywords we might suggest the use of PERHAPS to allow processing with information which is not fully true or fully false:

```
IF (PERHAPS this might be true)
```

```
THEN
```

```
  (TRY such and such a goal)
```

which can be read as:

```
"if it might be the case that ... then try effecting the goal
whose content is ..."
```

This allows a sense of degree of belief without the complexity of Bayesian values.

A third strategy is to allow the rules in the knowledge base to be compartmentalized. We might invoke a block of rules by using a DO keyword:

```
IF (conditions warrant it)
THEN
    (DO applicant has child rules)
```

There are several advantages which might accrue from this. First, the context problem [Goldstein, 1977] is to an extent overcome - we can treat these blocks of rules as we might treat subroutines or procedures by specifying which pieces of information can be passed to the rules and what the results being passed out are. Thus, we will not be concerned about the triggering of rules in other parts of the knowledge base or other blocks. The context problem arises either because the knowledge base is too large or we wish to change the view which we have - at one point we ask questions about the applicant, perhaps, and next about their parents or dependants/lodgers etc. With complex changing viewpoints, it is very difficult to ensure that side effects from triggering rules do not cause undesired effecting of other goals.

The second advantage comes from being able to treat a large knowledge base as smaller blocks of rules - the rules are easier to handle, remember, edit and suchlike; this advantage should not be underestimated when the number of rules in a knowledge base gets over 100, say.

One of the problems discovered with the present system is that concerning easily updating the legislative rules held in the system. This is important in legal systems; for example, since the ELI rule base was set up, the legislation has been amended on several occasions and also new rulings have been made by the Commissioners and the Secretary



of State. Currently ELI only allows searching of the rule base by string handling means - just as a word processor operates. While this is useful, we wonder whether it is ideal, or whether some means of "conceptually indexing" the rules in the knowledge base should be used with more advanced tools (such as [Reboh, 1981]). In many ways, this problem is similar to that problem of updating some data bases. No approach used there has shown itself to be overtly superior, but one possible idea is to use the notion of describing the general area by means of a classificatory tree (as used by Dewey indexes in librarian's systems) and indexing individual rules according to this.

The problem of "losing" individual rules within large knowledge bases is perhaps a problem which will appear more frequently in legal IKBS systems due to the expected size of their knowledge bases. It is therefore, most certainly an area of future research.

#### 8.2.2 Query By Assertion And Explanation Facilities

Much has been made of the abilities of expert systems to allow explanation and to allow query by assertion. The ELI program currently contains program stubs which can be easily extended to allow these facilities.

We do not however see these as being essential requisites of the system at the present time. We feel that although easy to add, we are not quite sure what people who would interact with the system actually require. It has been stated by Slayton that legal retrieval systems should be designed to accord with the working practices of lawyers:

What is needed is a pause in funding and development while emphasis is given to serious study with four objectives:

1. research into the nature of legal thought processes;
2. careful experimentation with operating retrieval systems to determine their exact capabilities and to compare their results with those of manual searching;
3. careful juxtaposition of conclusions concerning thought processes with the constraints imposed by, and the results of, electronic retrieval; and
4. development of the experimentation with advanced artificial intelligence systems (such as the McCarty TAXMAN project) to determine whether and in what respects they constitute a line of development worthier of pursuit than development of established retrieval systems."

[Slayton, 1974]

Slayton's remarks caused some controversy in Jurimetrics Journal at the time, but discussion with lawyers has led me to believe that there is a distinct difference between what, for example, these lawyers would like and what they actually receive from legal retrieval systems. Therefore, it seems sensible to spend some time, in the development of the ELI system, to ascertain the requirements of the "para-legal" users of these programs rather than simply to provide facilities that - as a computer scientist - we think is needed. Below we deal with this further.

#### 8.2.3 Utilizing An Algorithmic Language

There are two main reasons why INTERLISP is not the most ideal language for a system which is seen as a developmental vehicle. They are a) speed and b) portability.

We have already mentioned some of the problems of speed - the use of EQUAL as opposed to EQ just being one of the problems - but a second problem is that of portability. Put quite simply, there are so many dialects of LISP, and so few implementations of INTERLISP that it is difficult to allow potential users of such programmed systems to use them.

Obviously, a portable language is a desirable feature. It will allow users to use the system as an "expert system shell", as a research aid, and (as suggested by one lecturer on a course on "Computing and Law") a teaching aid. The only problem is deciding which language to use; that decision will be made later, but since the ELI system requires access to the interpretation element of the production system it is unlikely to be one of those languages which provide a ready-made interpretation facility (e.g. PROLOG and suchlike). Also, since it is difficult to justify using as similarly as esoteric a language as INTERLISP as a tool for reasons of portability, it seems sensible to use one of the commonly available and used programming languages - say, Pascal.

#### 8.2.4 Revise The SETUP Facility

The SETUP facility was originally conceived as simply an editing and rule writing module, but it is becoming obvious that its role might well be increased. Particularly important would be the extraction of the conceptual matching module from the BUILD module, and the implementation of this within the SETUP module.

One of the main reasons for this is that such a substantial amount of processing time is taken up by conceptual matching within the BUILD procedure, that the BUILD process takes a substantially large amount of processing time (particularly when the number of rules being incorporated is over, say, 100). By putting this processing in the SETUP module, it could be carried out much more efficiently, and

probably much more elegantly.

The SETUP module could also have the facility to carry out the indexing of rules mentioned above. Doing this as the rules are extracted from the legislation etc. should make it convenient for the expert.

#### 8.2.5 Conclusion

We reiterate that the ELI program is not seen as a commercially implementable system; useful legal expert systems (outside perhaps of small, legally uncontroversial areas - perhaps, advising on the "easy" aspects of House Conveyancing while leaving complex cases to the expert) are still some distance off. Also, since this project has been carried out by one person working alone, the programming solutions provided are only one person's solutions - we do not see that they are necessarily of any greater effect than other possible solutions.

However, we do feel that the project has a substantial amount of "research space" open to it, and that the indications of future extensions mentioned above are only a small proportion of possible extensions.

### 8.3 MAN/MACHINE CONSIDERATIONS

#### 8.3.1 Lawyer/machine Interaction

In IKBS research we currently face the problem of being able to design useful problem-solving programs with high levels of expertise in domains where that expertise is rare and expensive while being unable to persuade all their potential users of their utility. For example, the MYCIN system although available to doctors in Stanford Medical School is not used by those for whom it was intended. A listing in [Bond, p179, 1981] illustrates how few current expert systems are actually being used by their proposed recipients, and the point has also been made by

Woolgar:

"An initial surprise in surveying the expert systems literature is the marked discrepancy in available reports about the state of research in expert systems. In addition to the extraordinary optimism of some reports, there is considerable caution, not to say pessimism, about the achievements to date. On the one hand, expert systems is generally regarded as one of the most active and exciting areas of AI research. On the other, there is considerable concern about the fact that the field currently faces "fundamental problems". In terms of the number of expert systems in existence, we can find claims that "nearly fifty" had been built by early 1982. In the less popular press, however, it is admitted that despite impressive performances by some of these systems, only four of the best known systems are in regular use. According to some respondents, even this is an overestimate!"

[Woolgar, 1984]

Woolgar references [Davis, 1982], [Duda, 1983] and an interview with Szolovits at MIT to support the point.

The MYCIN system was designed with substantial input from a medical team and the design work was carried out in a medical school. Not only were these positive environments available, but as Shortliffe states:

"MYCIN has also been developed with more attention to human engineering than is typical of much of the AI field. The goal has been to develop mechanisms for interacting with medical professionals who are not only unfamiliar with AI but have never used computers before. MYCIN's rules have therefore served as a highly useful representation scheme since they can be individually retrieved in order to explain why questions have been asked or to justify aspects of the program's advice. As AI applications for

use by scientists and other individuals become more common, MYCIN may well suggest some useful guidelines for interactions with novice computer users."

[Shortliffe, p376, 1975]

Given such input by potential users of MYCIN and such an optimistic view of its "user-friendliness", the question remains - why is the system not currently being used by the doctors for whom it was designed?

We can examine several possible reasons: legal, usefulness, convenience, complexity which can all be seen to be relevant to both medical usage and legal usage of computers.

#### 8.3.1.1 Legal Problems -

It is perhaps ironic that, in discussing the use of computers in the law, the reverse must also be considered. The computer since its inception, though, has caused legal problems, from the problem of the copyright of software [Tapper, 1973] to the use of computer output as evidence in court [Sizer, 1982]. Many of the problems for IKBS programs have arisen because

##### a) computer programs are error-prone:

"For several years now there has been considerable disquiet expressed about the problems of designing and producing large software systems. This concern has even been called a crisis and has resulted in the use of the term 'software engineering'. It is said that there is little software which is reliable and which is based on the best techniques currently available. ...

Reliability should be thought of as a relative term, being related in some way to the application under consideration. In applications such as air traffic control the computer system must never fail; in other cases, some level of performance slightly short of this may be acceptable."

[McGettrick, 1982]

b) their operation is opaque to the user. In fact, their operation is even opaque to other 'maintenance' programmers:

"A fundamental problem with program maintenance is that when a change is made it often introduces unforeseen side effects. Fixing a bug has a substantial chance of introducing a new bug.

Often a change has system-wide ramifications which are not obvious ... This situation is made worse because the maintainer or repairer is often not the person who wrote the original code ... Maintenance changes tend to deteriorate the structure of programs, often making them more complex and more difficult to maintain next time."

[Martin, p8, 1983]

The former problem is one which has haunted computer scientists from the earliest days of programming and turns up in all situations where programming is used. The latter problem arises in part from the former (that is, the user has to take on trust that the program does what it is supposed to do) and also from the fact that computers are more and more being used by people who have very little real conception of their operation or construction.

The particular problems which accrue when IKBS programs are used is that the user is expected to "know" less than the machine. While this is satisfactory when the system produces reliable and correct advice, it is

not satisfactory when unreliable and incorrect advice is produced.

It has been suggested by various researchers that providing explanation capabilities within IKBS programs will overcome this problem of opaqueness. While this does have an intuitive appeal, we feel that explanation capabilities are most valuable to those who might already understand the program's operation (ie its sphere of knowledge); they will not be properly comprehensible by those with a naive or novice view of the problem domain. There is, of course, no real evidence to believe that the problem of opaqueness has been completely overcome by the use of the expert system methodology; perhaps the available evidence argues against such a view. For example, MYCIN's knowledge base was so tender to rule input that a committee was set up to decide which rules were to become part of the generally available knowledge base of the system: .

"Although we are eager to permit experts to teach the system new rules, there are potential dangers in letting anyone have uncontrolled access to MYCIN's knowledge base. ... We therefore do not yet automatically store new rules as part of the permanent Consultation System. Instead they are stored temporarily in a file assigned specifically to the expert from whom the rules were acquired. Whenever that expert uses the system he may load his personal files and they are automatically added to MYCIN's knowledge base. MYCIN project members have an opportunity to examine both the new rules and the English text from which they were derived, however, before the new knowledge is transferred from the expert's personal file to the permanent Consultation System."

[Shortliffe, p344/5, 1975]

Shortliffe briefly dealt with the legal problems, such considerations being important in the U.S.A.:

"For example, some physicians may be reluctant to consult the



program until they know the legal ramifications of following or ignoring MYCIN's advice. Hospital lawyers may be able to provide assistance with such questions. If there has been any test cases on the subject, however, they have not been nationally prominent, and it is therefore difficult to state with certainty who must accept responsibility. I have stressed, however, that MYCIN is a tool for the physician and not a replacement for his own clinical judgement. It therefore seems likely that the ultimate legal responsibility will rest with the clinician rather than with the computer system or its developers."

[Shortliffe, p333, 1975]

Campbell has noted that the professional (who will interact with these machines) is expected, by legal principle, to:

"maintain a competent knowledge about the tools and standards of his profession [Rogers, 1979], and that he should not neglect to make use of state-of-the-art technology in his field there this a means of avoiding injury or damage [L.T., 1945]."

[Campbell, 1982]

Campbell also writes of the particular problems for the legal professional inherent in IKBS systems:

"Firstly, expert systems set expert knowledge into a framework that may be qualitatively different, as experienced by the trainee professional, from the framework in which his traditional education is conducted. Therefore, unless he has had a significant exposure to methods of diagnosis or reasoning in his subject that are explicitly tied to rules of the forms R1 or R2 (rather than conducted by analogy or example, which is normal in medical and some other professions), he may (i) have no feel for what is behind a given output from an expert system, or (ii) be unable to

formulate questions whose answers are essential for the prevention of an injury and which an expert system may be fully able to provide, or (iii) fail to understand that a particular publicized enhancement of a rule-base allows certain expert systems to solve new classes of problems in his field, or (iv) appreciate that proprietary system A to which he has bought access cannot solve a class of problems that a rival system B can handle because of its state-of-the-art features. These problems contain some messages for deans of medical and other professional schools, in connection with planning and curricula."

[Campbell, p9, 1982]

We do not know how such legal problems might be resolved; but such is the situation with the problems created by so much of today's new technologies. Presently though, the legal position is unclear about who is financially liable for damages from a system which produces incorrect advice; it may be possible to sue the computer scientist just as easily as the legal user of the system.

Concluding then, there are many legal problems which relate specifically to the use of computers and the introduction of computers to areas which were previously free from "programming". The area of legal knowledge engineering is but one more.

#### 8.3.1.2 Persuading Potential Users -

Shortliffe has seen the problems of the acceptance of MYCIN as being related to the educational needs of the medical profession, and that educational programmes:

"demonstrate that physicians will learn to use computers and accept their role, if the benefits of the technology outweigh the costs of learning to use the devices and the costs of integrating them into one's normal routine."

[Shortliffe, p17, 1982].

Teach and Shortliffe reported upon the effects of a two day tutorial used as a means of testing the "before" and "after" views of doctors concerning medical computing systems. It is interesting that the physicians on the course were not found to be generally opposed to the use of computers on the ward, but were to those which appropriated their own responsibility for patient care:

"Applications that were presented as aids to clinical practice were more readily accepted than those that involved the automation of clinical activities traditionally performed by the physicians themselves. The distinction between a clinical aid and a replacement seems to be important to physicians and suggests design criteria and preferred modes for the introduction of computing innovations. This perspective is consistent with historical attitudes regarding the adoption of other kinds of technological innovation. For example, computerized axial tomography has been widely accepted largely because it functions as a remarkable useful clinical tool, providing physicians with faster and more reliable information, but in no way infringes on the physician's patient-management role."

In many situations it is possible to overcome the opposition to new technology which appropriates a job skill (for instance on the factory floor), but there are as Nickerson has written other groups:

"Some computer systems are used because their users have no choice in the matter. That is to say the users must use the system or they cannot perform their jobs. Examples include airline reservation systems, hotel reservation systems and sales registration systems. My interest ... is in systems that do not have to be used and in users, or potential users, who are not required to be users by virtue of their jobs. This delimitation of the problem will have the effect of focusing attention primarily on people, such as managers and professionals, who tend to have considerable latitude in how they perform their jobs, as opposed to technicians and clerical personnel whose tasks are likely to be more tightly prescribed."

[Nickerson, p469/70, 1981]

Perhaps one of the most persistently used IKBS program of the current generation is the DENDRAL system [Lindsay, 1980]. This system can be seen in this "technical aid" light, it carrying out a large amount of tedious processing which had to be done by hand previously. The DENDRAL system therefore aided the user without detracting from the user's control over what should be done with the results.

Similarly, the introduction of legal information retrieval systems was carried out without detracting from the managerial or decision making role of the lawyer. Having searched through one of these system's data bases, it is still required of the lawyer to analyze it.

But in legal knowledge engineering, the question arises: what if no lawyer wishes to use a system which actually tells him how to go about his daily work? Because, that is the very goal of IKBS programs - they extract the expertise from one person and give it to another, in the manner that an apprentice must accept the expertise of his or her craftsman.

It is currently held by Teach and Shortliffe that the answer to this problem is - in the medical field, anyhow - to:

"Strive to minimize changes to current clinical practices. The system should ideally replace some current clinical function, thereby avoiding the need for an additive time commitment by the physician. The system should ideally be available when and where physicians customarily make decisions."

[Teach, p556, 1981]

There have been several descriptions of the "legal decision process" written; it has been noted that:

"electronic legal retrieval systems designed to assist in, or substitute for, a key part of the legal thought process have been developed with little understanding of what the process is, and what the consequences of changing it will be".

[Slayton, p21, 1974].

It seems that this approach has succeeded with legal retrieval systems despite this lack of analysis (but see some of the problems alluded to in [Larson, 1980]); but we should not expect it to occur with IKBS programs.

What we are implicitly saying is that the future research of the ELI system cannot suffice with purely computer science goals (that is, more effective representation, easier and quicker access of that represented information) but that we should note the point often made by the recipients of computing technology - that computer scientists should design what they need rather than what is perceived to be the need. In order to do this, we believe it is essential that the system should become the research focus of both computer scientists and lawyers, rather than the situation at present (where only the author has had any say in the design).

We do see a time lag between these first experiments in legal knowledge engineering and the eventual (useful rather than novelty) implementation of such systems in less than trivial domains; in this time period we see the ELI system undergoing many changes in design specifically to incorporate the needs of the legal profession. For, after all, we can justify this process in such a complex area as the law by recourse to Buchanan:

"However, there is no reason to believe that the expert's first conceptualization is complete or correct. The expert will begin with classical textbook statements that are reasonably certain and well accepted. But in domains like mass spectrometry and medicine there is still a gap between what the textbooks say and what is needed for expert performance. ... Similarly, there is no reason to believe that the programmer's first implementation is correct and appropriate. There is tremendous potential for misinterpreting what the expert says when the programmer has little understanding of the domain. A more serious problem is that the internal representation chosen by the programmer, on the basis of initial conversations with the expert, will be appropriate for ideas to be incorporated later. ... Thus it is not surprising that new applications programs often need to be completely rewritten after

their initial implementation."

[Buchanan, p418, 1979]

Accepting that we do not see the current ELI program as the final solution, though, the important problems of lawyer/machine interaction to be solved will relate to the following:

- a) who will use the IKBS program
- b) why will this person want to use the program
- c) where in the user's workload is the system to be directed
- d) what specific representational problems will this user have - that is, what is his/her view of "the law" and need it be integrated into the system
- e) how can the usefulness of the system be quantified
- f) how can the use of the system be improved

Of course, there may well be more than one type of user; therefore, in each case the above points will need to be investigated.

## CHAPTER 9

### THE LIMITS TO RULES AND LOGIC IN LEGAL KNOWLEDGE ENGINEERING

"Logic, Law and Switzers can be hired to fight on anybody's side."

16th Century Proverb

#### 9.1 INTRODUCTION

There is a substantial debate in certain areas of computing research on the possibility of intelligence being rule-based. The view is prevalent in cognitive psychology and artificial intelligence as we have mentioned earlier. This view has also been seen in jurisprudence as we described earlier when providing a model of Neil MacCormick's thesis. In mathematics we might suggest that the debate is about the search for a decision procedure or means of effective computability. It is my contention that these three views - I shall call them all formalist - are representative of the same error in epistemology. That error is one of failure to see that human understanding and constructions are the outcome of an essentially social process: life, we can say, is not effectively computable. And understanding that error, if accepted as such, leads to a position which can be put quite clearly as:

the legal process cannot be captured by formal methods

This means that none of the attempts which have been made so far to represent "the law" will be successful. No computer technique or logical system has the capacity to describe the legal process. Such a perspective is controversial and needs to be argued. In this chapter I



present this argument and suggest that this argument offers legal logic programmers a clear statement of a position which is contradictory to the one which they provide: if they accept my argument, then they cannot accept the utility of formalizing the law.

Since the research on the ELI project began, the same rule based approach has been utilized by other researchers. We might almost say that it is becoming a commonplace to attempt to incorporate "legal rules" into legal expert systems (or at least to consider doing it). As examples of this ongoing research, we can point to two other projects which have used, to some extent or other, rules as the primary entity in their knowledge bases (i.e. [Cory, 1984] and [Waterman, 1981]). Since the work on the ELI project began, difficulties and problems have arisen which suggest to me that rule based methods are of limited use in expert systems which attempt to provide advice built upon models of "rule orientated legal reasoning". We posit that this limitation is a direct result of the incompatibility between logic and the legal process - put simply, "law is not logical". It seems that other researchers, too, are becoming disillusioned with these methods, but unsure from where they arise or how they can be solved (for example the personal note from a member of the Alvey/DHSS Demonstrator project [Lapham, 1985]); in this chapter we hope to clarify the problem and pose limitations on the solution to that problem.

When the ELI project originally began, it seemed obvious that the law and legislation could be represented as relatively simple (even if sometimes verbose) rules. Implicit in this belief was a following of that school of jurisprudence termed legal positivism, a school which posits the legal rule, rule interpretation and rule following as the principle elements of the legal process.

By capturing enough of these rules from legislation, case reports or secondary legal sources, it was thought possible to represent "the law". The capturing was to be done by the expert. This means that we did not see the legislation as the only basis for the expertise - the encapturing of the rules was to be done by the expert interpreting the legislation, together with any information from secondary sources, case reports or indeed, the personal opinion of the expert.

Given such a potentially large number of concepts and rules, it might seem as though the major research problems would be in the physical handling and storage of the rules - in other words, research would become orientated towards solving computer science problems. As examples of these problems, imagine trying to ensure that the rules triggered in the correct manner (a problem known as truth maintenance of the knowledge base) when the rule corpus became as large as, say, two thousand rules. Another problem was seen to be the need to prune the corpus as the interaction between lawyer and system took place - obviously we do not want to test rules which have little relation to the problem in hand since this only increases the time of interaction and boredom of the users. Other computing problems are seen to be of paramount importance if we accept that it is valid to represent the law as a collection of rules; we gradually move the research question from "what are we trying to encapsulate" to "how do we manipulate what is encapsulated".

The distinction between the two questions is most decidedly important - with the first question we state that we are unsure about first, what the law is and second, how it might be represented in a computer program. The second question states that we can ignore any jurisprudential aspects of the research area and concentrate upon the computer science aspects.

In Chapter 5, above, we suggested that at least one project had approached the field of legal knowledge engineering in what was a grossly simplistic way. Their approach was directed towards the second research question, and they felt that any legal or jurisprudential aspects might be safely ignored. The research for the ELI project is novel in that we have always attempted to handle both of these questions concurrently - developing a rule based expert system on one hand, and trying to come to terms with just what it was that "the law" is. Since we felt, and still feel, that commercial legal expert systems are some distance from general applicability, this particular research methodology was considered essential. Here, we shall try to enunciate some of the conclusions - but primarily those of a negative nature - which have been reached. Conclusions which have not emanated from other IKBS legal research. It should be noted that this position is novel in the drawing together of the philosophy of law, philosophy of logic and computer science; and also, it is novel as a pointer to the most appropriate kind of legal knowledge engineering which might be achieved given - as we argue - the limitations of rule based approaches. We suggest that the novelty of this position is more than superficial; it offers a radical new insight into the potential of the research field. It is the only statement from a researcher in the field of legal knowledge engineering which, so to speak, swims against the tide. That tide is epitomized by the view that by the refinement of our tools - whether conceptual or actual - we will some day have systems which can provide answers to legal questions. I swim against this orthodox view and present the heterodox perspective that no computer tools or methods will ever have the capacity to represent "the law" or "answer legal questions". The novelty of my position is, in part then, that I propose the most limiting constraints upon the research field yet stated.

It is important to recognize also that this position was not one which was taken as an a priori presumption; it grew out of the research problems cast up from the attempt to marry computing and law. Such questions as, what is law, how does legislation stand in relation to language to logic, what are the "intelligent" aspects of IKBS techniques, were those which led to the conclusions dealt with in the remainder of this chapter.

I shall start by examining the view of the legal rule which I took when beginning the research, then how this was incorporated into the ELI program. After that, I shall present a criticism of the very notion of the validity of "legal rules" as concrete entities which can be handled by legal expert systems. Then a return to Prolog and "real law" which should be seen as a challenge set to the orthodoxy from the heterodoxy.

## 9.2 CLEAR RULES

It has long been held that the very notion of just what a legal rule is is problematical. For instance we can see the reference to them by Twining and Miers:

"We are now sailing near some very deep waters. What is involved in 'understanding' a situation, a rule, or the law is a central, and extremely problematic, question of social theory. In studying interpretation of rules we are here faced with a dilemma: we cannot reasonably expect to plumb the depths of philosophical questions about the nature of understanding social phenomena, but neither can we pretend that such questions are irrelevant."

[Twining, p167/8, 1982]

And we can also see the more neo-Platonic rules of H.L.A. Hart, rules which he classifies as primary and secondary. We can term them neo-Platonic rules because Hart's rules seem to possess an ontology

which is separate from the creators or enforcers of the rule. For example, Hart writes:

"At any given moment the life of any society which lives by rules, legal or not, is likely to consist in a tension between those who, on the one hand, accept and voluntarily co-operate in maintaining the rules, and to see their own and other persons' behaviour in terms of the rules, and those who, on the other hand, reject the rules and attend to them only from the external point of view as a sign of possible punishment. One of the difficulties facing any legal theory anxious to do justice to the complexity of the facts is to remember the presence of both these points of view and not to define them out of existence. Perhaps all our criticisms of the predictive theory of obligation may be best summarized as the accusation that this is what it does to the internal aspect of obligatory rules."

[Hart, p88, 1961]

The reader of Hart's "Concept of Law" will look hard to find a full definition of the qualities of these rules (he states [Hart, p8, 1961], for example, "What does it mean to say that a rule exists?" but fails to provide a non-circular answer) but will be aware from reading through the text that, to Hart, they seem to have, what we might call, "a life of their own" - the rules are the objects which control human behaviour: rules which people follow or circumvent. But although the conjunction of these primary and secondary rules are, to Hart, the essence of law there is a further clarification - law has an "open texture". This open texture is an answer to the rule-sceptic who would accuse Hart of formalism:

"In fact all systems, in different ways, compromise between two

social needs: the need for certain rules which can, over great areas of conduct, safely be applied by private individuals to themselves without fresh official guidance or weighing up of social issues, and the need to leave open, for later settlement by an informed, official choice, issues which can only be properly appreciated and settled when they arise in a concrete case."

[Hart, p127, 1961]

The legal system, then, is composed of rules which classify, but carry this out to an extent depending upon the fact situation to which it is applied. As Hart claims:

"All rules involve recognizing or classifying particular cases as instances of general terms, and in the case of everything which we are prepared to call a rule it is possible to distinguish clear central cases, where it certainly applies and others where there are reasons for both asserting and denying that it applies. Nothing can eliminate this duality of a core of certainty and a penumbra of doubt when we are engaged in bringing particular situations under general rules."

[Hart, p119, 1961]

A successor to, and indeed refiner of, Hart's thesis is Neil MacCormick who, as we stated in Chapter 4, presents a model of judicial behaviour which is both analytic and normative - it attempts to describe how the judiciary arrive at a decision through a deductive procedure (with legal rules as the elements of the deductive form) and also upholds this as the correct manner in which legal rules should be applied. Thus MacCormick poses strict logic as the method of legal reasoning. He states:

"Scotsmen and Scots lawyers by contrast [with English lawyers] have taken some pride in being logical and in having a legal system

which exhibits the virtues of logic.

.... The Scottish system, and indeed the Civilian systems generally, are less likely to be subject to that mistake [of using observations which might be true in the non-technical logical sense but which are false in the technical logical sense]; but perhaps such systems are at risk of suffering from the converse equivocation. Since legal reasoning is a form of thought it must be logical, i.e. must conform to the laws of logic, on pain of being irrational and self-contradictory. That is, law must be 'logical' in the technical sense."

[MacCormick, p40/1, 1978]

Generally, then, the framework within which I operated was that of "the Oxford school" of legal positivism, a school which claimed primacy for rules and their technical application (by which I mean that their prescription for how the law ought to operate was technical - the judge should attempt to apply the rules in as logical and strict a manner as possible - there should be no recourse to any notions of, for example, "justice". This justice arises, they would claim, from strict application of the legal rules). This framework is the general view held in British and American (and perhaps other) jurisprudential thought; it is a framework which it is natural for a computer scientist to initially accept.

One of the most basic assumptions of the Oxford school is that there is such a thing as a "clear rule" - it is a rule which can, to a large extent, be applied without further thought. It is, to Hart, the core of certainty which "nothing can eliminate"; to MacCormick it is the major premise from which the judicial argument must and does proceed. Later in this chapter I shall suggest that the very idea of a clear rule of law is an invalid idea, and that it cannot be used successfully to provide legal expert systems which can predict real judicial decisions;

in the next section we shall see how the concept of a clear rule was encapsulated in the ELI program.

### 9.3 CLEAR RULES IN ELI

First we must point out that when we speak of a clear rule here we do not necessarily mean a logical sentence (though we might); rather we are usually speaking about a piece of text which might run as:

"Read the rest of this text"

Others might wish to formalize such a sentence into the predicate calculus or deontic logic, but this was never our aim.

What is important about clear rules is that they can be written and interpreted with the minimum of confusion and that, to an extent, they almost (but not quite) "step forward and claim their own context". Also, and this is correlated to the lack of confusion, they are held to be commonly the case: these are the rules of law over which there is no or little argument. It is only in what are termed "hard cases" that there is a problem of interpretation (involving the penumbra of doubt); in the easy cases the core of certainty is to the fore.

It was relatively easy to describe these clear rules (or rather, the core of certainty of these rules) by simply handling them as sentences which either singly or in multiplicity make up an antecedent or consequent.

The traditional production system interpretative strategy was not considered suitable for interpreting these rules. Firstly, because there would be much complexity in positioning individual rules within the listing of productions (since in the traditional method interpretation begins at the head of the list, incrementing down until one rule is triggered and then restarting at the head of the list again). An implicit context is thus awarded to each rule in the corpus



which, unfortunately, requires a large amount of manipulation from the builder of the knowledge base. In fact, any architecture will have a context problem, but we used a hierarchical structure to try and overcome some of these problems (we do see, by the way, that a more full investigation of that structure is required; and would hope to carry it out in the future).

But what of the "penumbra of doubt"? This was provided by attaching annotations to the individual rules or parts of rules. Annotations might typically cover aspects such as where the rule was derived from (Act, Section etc. or secondary legal source) or a fuller explanation of what the rule might mean or any qualifications which might be added. The user, then, when asked whether a certain rule element is true or not could call up the (i) annotation linked to that rule of which that antecedent was a part, and (ii) the annotation linked to that antecedent. Thus an element of doubt can be incorporated - the intuition of the user can be left as the final decider.

Hart's model was of a two part rule system - primary and secondary rules. Since the ELI system was not attempting to model the entire legal system it did not require "secondary rules [which] are all concerned with the primary rules themselves [as] they specify the ways in which the primary rules may be conclusively ascertained, introduced, eliminated, varied, and the fact of their violation conclusively determined." [Hart, p92, 1961]. I therefore did not include any such refinement.

I have attempted to give the flavour of the ELI program here with specific reference to its closeness to a particular school of jurisprudence, not to fully describe it; of course, earlier parts of this text give more details. In the next section, we present a criticism of that jurisprudential school, important since that school offers the orthodox view of legal interpretation and prediction.

#### 9.4 A CRITIQUE OF CLEAR RULES

Hart and MacCormick posit that we can view the law as a collection of rules which seem, at least to me, to have 'an existence of their own'. It is not unfair to see these two writers as agents of a formal or technical view of the law. It is also not unfair (especially when we remember Whitehead's statement that, "The safest general characterisation of the European philosophical tradition is that it consists of a series of footnotes to Plato") to suggest that this formalism is of a neo-Platonic nature; the rules are seen to be set apart from any human participation in their invention and in their application. Of course, the individual rules may be amended (Hart speaks of secondary "rules of change") but the outline, "idealized" form of a legal rule remains. Analogously, the triangle may have different dimensions, but the Platonic geometer will still have a purer conception of "triangle" than the individual example. I do not use the descriptor "neo-Platonic" as a positive adjective, since I see it as an attempt to avert my eyes from real life; Bertrand Russell suggested that the division between Platonism and non-Platonism seemed to be a division between those who preferred perfection to real life and those who preferred real life to perfection.

The problem I see with this view of clear legal rules is that it ignores the social determinators of the process: by presenting a technical methodology as to how the judiciary ought to reason they present an essentially conservative view (a point which I take up when looking at the Prolog project again, below) which bears little relation to empirical evidence. By empirical evidence, I mean the written case reports - the substantive law - which present a completely different story. By looking at how the judiciary actually adjudicate, we can see that there is very little evidence for the notion of a clear rule which they can apply without further thought.

My first argument is borrowed from Moles [Moles, 1985] critique of Hart and restatement of Austin and Aquinas. Later I develop my own argument pertaining to the British Nationality Act (1981). Briefly expressed, Moles suggests that we cannot look at "the rule" in isolation - we can only see its existence within a social context since it depends upon its interpretation for the context in which it is to be applied. I shall later return to just what this means for legal rule-based expert systems, for now I shall try to concisely re-present his argument.

Moles takes a piece of legislation which might on first appearances appear to be the statement of a clear legal rule. The legislation (for those who might wish to formalize it) is a provision from the Domestic Violence and Matrimonial Proceedings Act 1976 S1 (1):

- S1 (1) Without prejudice to the jurisdiction of the High Court, on an application by a party to a marriage a county court shall have jurisdiction to grant an injunction containing one or more of the following provisions, namely, -
- (a) a provision restraining the other party to the marriage from molesting the applicant;
  - (b) a provision restraining the other party from molesting a child living with the applicant;
  - (c) a provision excluding the other party from the matrimonial home or a part of the matrimonial home or from a specified area in which the matrimonial home is included;
  - (d) a provision requiring the other party to permit the applicant to enter and remain in the matrimonial home or a part of the matrimonial home; whether or not any other relief is sought in the proceedings.
- (2) Subsection (1) above shall apply to a man and a woman who are living with each other in the same household as husband and wife as it applies to the parties to a marriage and any reference to the matrimonial home shall be construed accordingly.

Now this seems, at least to me, to be a relatively clear piece of legislation. We might think of it as an example of Hart's secondary rule with a core of certainty and a penumbra of doubt. No doubt the legislators who drafted it too, would have considered it relatively clear. But did the judiciary think so? Unfortunately not, as Moles points out. The relevant cases in the argument are B. v B. [1978] 1 All E.R. 821, Catliff v Jenkins [1978] 1 All E.R. 836 and Davis v Johnson [1978] 1 All E.R. 841.

Moles first points to the case of B. v B. where the trial judge took the view that it was unreasonable for Mrs B to continue living with Mr B (Mrs B have left the joint home, of which Mr B was tenant, leaving the children there) and ordered Mr B to leave the home within 14 days. Mr B appealed, an appeal which was allowed, the decision being arrived at on the fundamental issue of the proper construction of S1. A summary of the judgements is:

(1) the judges all agreed upon what the main facts and points of law were

(2) the approach to the meaning of the legislation is technical - it does not concern itself much with the social conditions of the case - only with the meaning of the law

(3) that technical investigation of the law is mainly concerned with trying to fit S1 into other existing legislation (in fact primarily of property rights, and the jurisdictions of the High Court and County Court).

In effect the bench decided that the provisions above had very little real effect - that although there existed a clear rule, it was of little consequence since other clear rules negated it. This was markedly different from a second case which came before the Court of Appeal within a few days of the delivered judgement in B. v B.. Once again, in this second case, a women had left the joint home which was tenanted by the male partner.

In this second case, Cantliff v Johnson, the trial judge had ordered removal of the male from the home (adding power of arrest without warrant to the order). The Appeal judges overruled this order in a similar manner to that of B. v B. - that is, in a technical light without any real investigation of the social facts. Yet, whilst the first appeal judges had interpreted S1 to have little real effect, in

Cantliff v Jenkins they found that it did in fact have effect - it was only effective against a person with no right or interest in the property, they decided, else it would have a transfer of property effect. And such a situation, they seemed to imply, could not possibly be.

Thus, we can pause here, and note that so far the provision has been interpreted in two separate cases in two separate ways. The judicial interpretations have, I would contend, little relationships to my view - as a layman let it be admitted - of what the provision is meant to do.

The matter arose again within a month in the Court of Appeal, in the case of Davis v Johnson. Davis left the home with her two year old child and had Johnson ordered to leave. Johnson applied to have the order lifted (after B. v B. and Cantliff v Jenkins); it was and Davis returned to the woman's refuge. However, on this occasion the result was somewhat different, with the Court of Appeal rejecting (strongly, in fact: "I am afraid that the judges sitting in B v B must have misunderstood the law as it is applied in the Family Division") the decisions of B. v B. and Cantliff.

Rather than looking at the purely technical way in which the provision relates to other legislation and substantive law, the Court of Appeal in Davis v Johnson looked to the broader perspective and the social circumstances. In effect, they looked at the same provision as the two previous courts had, but discovered that there was a different legal rule! Or, to be quite precise, four went for this different rule, and one stated a preference for that of B. v B.

We can present an empirical account:

There is first the layman's interpretation - "a", we shall say - of the rule (we shall give it a count of 1 - a conservative figure).

Then comes the case of B. v B. - "b" - where a different interpretation of the legal rule arises (we shall give it a count of 3, one for each of the judges on the bench).

Then Cantliff - "c" - where another interpretation arises (once again a count of 3 for the unanimous members of the bench).

Then Davis - "d" - where a specially constituted Court of Appeal of five produce two rules (4 for "d" and 1 for interpretation "b").

In total then, our "clear rule of law" has a scorecard of:

| <u>Interpretation</u> | <u>Score</u> |
|-----------------------|--------------|
| a                     | 1            |
| b                     | 4            |
| c                     | 3            |
| d                     | 4            |

So where, I might ask, is the core of certainty and the penumbra of doubt which we might expect to find? Cynically, I might suggest that there is a large penumbra of doubt and no core of certainty.

Now one argument which might be put against this is that, "Yes, clear rules are not always clear - but they are clear until the judiciary change them". In which case I would say, there is no difference between our positions - simply that you call a clear rule something which, to me, isn't clear! To me, the existence of the rule depends upon the personal view of the judge positing that rule - it is created, discussed, manipulated by the judge as an abstract concept. It is not pulled out from the law reports and legislation on that judge's bookshelf - it has no concrete life which leads to it being spotted automatically by any number of judges.

Another argument might say, "Yes, there are parts of the law which it is very difficult to formalize; we should therefore limit our attempts to find clear rules to certain sub-parts of the law. Perhaps tax law." This is of course an argument which is presented by McCarty when he

writes:

"it is tempting to start with the 'simplest' possible legal issues, such as the subject matter of the first-year law school courses. We might therefore be tempted to investigate assault and battery cases from the first-year torts course ..., or offer and acceptance cases from the first-year contracts course. But these cases are 'simple' for law students primarily because they draw upon ordinary human experience. .... Paradoxically, the cases that are most tractable for an artificial intelligence system are those cases, usually involving commercial and corporate matters, which a lawyer finds most complex."

[McCarty, 1980b]

My argument rests upon the assumption that there is no difference between any judicial interpretation of legal rules, whether they arise from tort, contract or commercial or corporate matters. The problem of finding the clear legal rule in any of these areas is insurmountable, my argument states, because there is no clear rule which cannot be overruled, forgotten or created by the judiciary. Take another example from Moles, an example which relates to tax cases, one of those cases which McCarty might consider most tractable. The case is Vestey v Inland Revenue Commissioners (nos 1 and 2) [1979] 3 All E.R. 976.

Vestey concerned access to a foreign-based trust and income tax due upon income from the fund. The crown claimed that where an individual had either received or had power to enjoy the income from such a fund, the person became liable to taxation on the whole of the income of that fund no matter how much had actually been paid. The previous ruling in Congreve v Inland Revenue Commissioners [1948] 1 All E.R. 948 had the unanimous support of the House of Lords upholding a unanimous Court of Appeal (a sure indication of a "clear rule") yet in Vestey it was overruled. Moles points to the importance of this decision:

"Thus change was called for, and if there was any doubt that the change was substantial, rather than marginal, it would probably be dispelled by the opening words of the article by Anthony Sumption in the British Tax Review:

"Congreve v Inland Revenue Commissioners had stood for nearly 32 years as one of the great landmarks in the development of the law enacted by [the relevant section of the Act] .... Now Vesty v Inland Revenue Commissioners (Nos 1 and 2) has come like an atom bomb to blow Congreve away and obliterate all the fortifications in defence of the Revenue which had been constructed upon the foundation of that case."

We all know that atom bombs are not used to make minor adjustments."

[Moles, 1985]

Yet another tax case, W.T Ramsey Ltd v Inland Revenue Commissioners [1981] 1 All E.R. 865 is dealt with by Moles. Briefly, Ramsey was concerned with tax avoidance, whereby a series of transactions were carried out in order to reduce a tax commitment. The previously held ruling ruled that each step in the whole transaction process must be seen in individual terms - if each individual transaction was legal, then the whole transaction was legal - that is, the tax avoidance scheme was legal. However, with Ramsey, the court took a decidedly different tack - they overruled that previous clear rule, and stated that it was now valid to look at the individual steps as being part of one larger transaction. Looked at this way, the tax avoidance scheme was considered illegal.

Moles strategy of actually looking to the weekly reports is an instruction which is worth more than the individual examples he cites (it is surprising how many philosophers of law manage to get by on a



handful of cases from the pre-war years). By taking up Moles' strategy, any number of useful examples can be found. But, of course, one requires the proper perspective in order to find the examples; a perspective which appears to be missing from Wasserstrom, when in the following he looks to the individual decision framed in logical terms, rather than the general pattern. Wasserstrom quotes the following from Fuller:

"The language of judicial decision is mainly the language of logic. And the logical method and form flatter that longing for certainty and for repose which is in every human mind. But certainty generally is illusion and repose is not the destiny of man. Behind the logical form lies a judgement as to the relative worth and importance of competing legislative grounds, often an inarticulate and unconscious judgement, it is true and yet the very root and nerve of the whole proceeding."

[Wasserstrom, 1961]

and then comments:

"Why, since common-law courts have been deciding cases for over 750 years, did the theory's inaccuracy remain hidden until the twentieth century? Again, the answer is soon found. If one were to look no further than the opinions that judges write to accompany their decisions, it would not occur to one that the decision process could be anything but deductive. For it is one of the curious features of Anglo-American case law that regardless of the way in which a given decision is actually reached, the judge apparently feels it necessary to make it appear that the decision was dictated by prior rules applied in accordance with canons of formal logic."

[Wasserstrom, p17, 1961]

We might suggest that the inaccuracy of the theory over the past eight centuries has not remained hidden. The confusion was clearly answered by John Austin in the Lectures [Austin, Vol3, 1863]; and the real confusion lies, as Moles suggests, with the Oxford School of Jurisprudence, and in particular the complete misunderstanding of John Austin's position by H.L.A. Hart (a misunderstanding now accepted, it should be noted, by Hart's defender, Neil MacCormick).

Of course, the cases we have borrowed from Moles and presented here have all been successful overrulings of previously "clear rules". Examination of case reports leads me to suggest that this rule breaking, manipulative process, is continual, though not always successful; many are the attempts of members of the judiciary to overrule previous decisions which are not successful (and this is related to my position that the very dichotomy of "easy cases" and "hard cases" is false). I can present one such case as the attempt by Lord Scarman (in Sidaway v Bethlem Royal Hospital Governors [1985] 1 All E.R. 643) to overrule the "Bolam Rule", a rule which holds that doctors, to act professionally, need only behave as a responsible number of their fellow professionals might reasonably behave in that situation. Scarman made specific reference to judicial decisions in certain States of the USA and in Canada; he also made specific reference to secondary materials. For instance an academic article, of which he said:

"This case, which has now been approved by the District of Columbia Appeal Court in Crain v Allison (1982) 443 A 2d 558, is discussed learnedly and lucidly in an article by Mr Gerald Robertson 'Informed Consent to Medical Treatment' (1981) )' LQR 102, on which I have drawn extensively in reaching my opinion in this appeal. The author deals so comprehensively with the American, Canadian and other countries' case law that I find it unnecessary to refer to any of the cases to which our attention has been drawn, interesting and instructive though they are, other than Canterbury v Spence and

Unfortunately, for Scarman LJ, his view of the rule was not accepted by other members of the Court, and that new and revised ruling did not come into being. But his comments tend to imply that Hart's secondary rules are not quite so slavishly followed, if at all, as he might imagine.

This critique, it should be noted, does not say that the concept of a legal rule is without use. I am not saying that we should not speak of legal rules. What I am saying is that we should be careful to remember that legal rules are objects (or rather "abstractions") of discourse, not objects with a concrete nature which we can mysteriously formalize and "find" in the legislation or the weekly law reports.

Before we turn to discussing just what this view of the judicial process means for legal expert systems, I might point out my view, which accords with that of Moles, on the technical application of rules which ignores the social situations of the receivers of judicial decisions. The early Appeal Courts made little mention of the physical factors which were involved in the cases; for example, in B. v B., Bridge L.J. related the facts:

"In the last year or two the relationship between these two parties has seriously deteriorated. There have undoubtedly been incidents of violence between them ... It is unnecessary to go into the matter at any length. In short, Mrs B's case was that Mr B had behaved so badly towards her that the relationship between them was at an end and there was now no prospect of reconciliation."

When, in Davis v Johnson the social factors were taken into account, we learn something of the the rationale behind the making of the legislation, a point noted by Denning M.R. about the previous decisions:

"The two decisions aroused consternation. Protests were made in responsible quarters. It was said that Parliament had clearly intended that these women should be protected; and that this court had flouted the intention of Parliament. So much concern was expressed that we have called together a full court - a court of all the talents - to review those two decisions; and, if satisfied that they were erroneous to correct them."

The social factors were less formally, yet more graphically, described by Denning M.R.:

"The judge said that there were two instances of extreme violence of a horrifying nature. On one occasion the man threatened her with a screwdriver. He said he would kill her and dump her in the river. He kept a chopper under the bed and threatened to chop her body up and put it into the deep freeze."

My point is, that we should be careful lest a desire for formality makes us forget that there is a world in which legislation and the legal process are of paramount importance. Below, we point out that some research in legal computing might tendentiously be described as of an extremely conservative and anti-democratic nature.

## 9.5 CLEAR RULES AND LOGIC IN LEGAL EXPERT SYSTEMS

### 9.5.1 Negotiation Not Clarity

Implicit in the critique of clear legal rules above was a view of what the legal process is. I will express it here, by enunciating its points.

First; there is no part of the law which is not constructed by social negotiation. Thus there can be no neo-Platonic legal ideals (be they rules, principles or whatnot) to which general legal recourse can be made, and therefore there must always be confusion over the meaning of

any legal idea or judgement.

Second; the role of a good lawyer is not to determine "the law" but to represent his or her client by persuasion of the judiciary involved in the case. The lawyer does this by presenting a rational argument: that is an argument which uses points which are considered valid in that court - in the same way that [Patterson, 1982] considered valid role behaviour - and which are constructed by trying to find previous judgements or "points" of law which might be interpreted in a favourable light to the client. Barristers do not accept any given rule of law (unless they wish to); rather they try to find ways of circumventing it by recourse to the particular facets of the case: the barrister is a rule-breaker. It is this factor which is most important in stating that rule based methods are not the most powerful ways to build legal expert systems; for the more powerful strategy will always be the rule breaking strategy. In the ELI system some attempt was made to overcome the lack of this strategy by use of annotations - but those can only clarify a rule, not suggest a new one which can readjust the interpretation of the knowledge base.

Third; the judiciary are not constrained by any notion of stare decisis if they do not want to be. We saw how Lord Scarman, for example, tried to overcome the views of previous judges by making recourse to an academic article about non-UK law. Even if the judiciary did want to apply strict canons of stare decisis (and there is no evidence to believe that they would wish to give up their "judicial creativity"), due to the very nature of things (the problem of language and personal viewpoint) they would be incapable of doing so in as full and consistent a manner as, say, MacCormick claims they do and ought to do. Few members of the judiciary would propose that "the law" is ever so simple that it can be applied in a formal way; this view of law says that we should not ask them to try.

Fourth; the application of the law can best be seen to operate in layers.

(i) At the lowest layer is the person who might have an issue which can best be adjudicated by law - since the legal process is so expensive this person will have little option but to forgoe proper solution to their problems. This is the group which must "accept the rules" provided by the legal system.

(ii) The next layer are those who can afford a proper legal solution to their problems - either by dint of wealth, or that their solution can be found in one of the cheaper courts (such as the Small Claims Court in the UK) or the state makes provision for their case to be heard (by Legal Aid or ex parte hearings in the UK) - and who thus have access to the legal process. This group has the ability to break the rules to a limited degree.

(iii) Third, is the layer which is - it sometimes appears - above the law, and who thus have a prime opportunity to make the "clear rules" of law even when opposed to those in group (ii). In this grouping we can place large and very wealthy corporations such as IBM, with their ability, for example, to outmanoeuvre anti-trust laws by putting a large number of high quality lawyers onto a case to apply delaying strategies (see for example [Malik, 1975] and his chapter entitled "A Lawsuit a Day keeps Justice at Bay"). Also, we can place large insurance companies in this bracket; they can choose the legal case upon which they decide to go to court - ensuring that the case has the best chance of success and that if successful, it will offer a legal ruling of future advantage to that company: many are the stories of insurance companies making out of court settlements in the final stages of a case - the payments are made to stop a new unwelcome ruling being made. The greatest degree of rule manipulation (excepting judicial

creativity) arises from this group.

Fifth; whilst the previous four are basically sociological, we must not forget the philosophical overview. We claim that the Austinian view of law as a sovereign power providing adjudication upon the meaning of legal rules is a much more sophisticated and correct view of the law than offered to us by Hart's "fresh start". Legal rules, to my perspective, have existence only in their enforcement; it is possible to discuss what a rule might be, but it is never possible to decide what a ruling will be until it is given and that action implies that power is available to enforce it.

I propose, then, that this view of law is not one which is consistent with computer systems based upon rule based methods because these systems present only rules - they cannot present rule breaking strategies. Neither, as a corollary, can they present rule supporting strategies, which are, in both criminal and civil litigation, a necessary opposition to the attempted rule breaking of the appellant's or defendant's counsel. The ELI system, judged by the standards of real law, must be a failure.

It should also be noted that this idea of a clear rule has far reaching implications, for we can see the same strategy involved in the logical description of law as we see in the use of clear rules. In fact, a logical sentence is just another form of clear rule - it contains no contextual information, it is a piece of law which is supposed to explain its own context, and which, its proponents have argued, is not open to negotiation. As we have argued earlier, the perspective that logic is of that nature is naive and incorrect.

#### 9.5.2 The Ouster Clause - A Problem For Formalization Of Law

It is our contention that logical methods are not appropriate for

handling the law, since the real problem of law is not the clarification of individual legal terms from the legislation, but the control of the judiciary. If we wish the judiciary to apply the "rules of law" with the purpose intentioned by the legislators, we cannot do so by giving them "clear" rules of law since they will simply use "common law" arguments to interpret these terms. And, since the common law exists only as a large collection of legal texts, commentaries and legislation it is not difficult to find supporting evidence for any position which the individual judge might wish to hold. Note, for example, how Lord Scarman in Sidaway went to an academic article concerning law in North America to bolster his argument: legal textbooks would not normally suggest this as a correct source of U.K. common law, but that didn't stop Lord Scarman.

Legal expert systems are, of course, involved with prediction of the judicial decision. And such prediction can only occur when there is a clear pattern of past events: some have suggested behavioural techniques [Schubert, 1963] and [Haines, 1922] has suggested that personal knowledge of the members of the bench is the best form of aid to prediction. Others have suggested that prediction can only occur when there is a clear formalization of the legislation which the judiciary cannot but fail to understand - they will be compelled by clarity to apply the rules in the desired manner.

The predictive method used by the ELI project was a weak form of the behavioural technique. We suggested that by using the interpretation of the lawyer as a means of building the rule corpus - and allowing any annotations that the expert thought appropriate - that the expert might, due to the very nature of being an expert, be able to weakly predict how the judiciary would interpret and apply the rules of law. We suggest now that, due to the very nature of legal rules, even this weak prediction is too weak: it is only useful as a means of discovering how a bureaucratic organisation might apply rules.



Formalizers of law, however, would argue with this suggestion. They would claim that strong prediction can be achieved from a proper normalization of the law. Clarity of legislation - they would claim - is the aim which is important. We wish to suggest in this section that the ELI project was closer in understanding the true situation regarding the need to interpret the legislation and then input it into an expert system rather than as, for example, the Prolog project claims is the requirement to simply normalize the legislation. We intend to do this by recourse to the idea of an "ouster clause". We shall outline what it is, give a brief description of the way it has been applied and then point out that one exists within the British Nationality Act (1981); the existence of this brings into substantial doubt, the claim to be able to formalize a piece of legislation.

### 9.5.3 The Ouster Clause

Administrative law is, broadly, the law relating to the control of governmental power. Thus this law deals with the powers which departments of a government have, and how they ought to be applied. The British Nationality Act (1981) would be described as one piece of administrative law, dealing as it does with the way in which the Home Secretary (or his agents) can dispense or deny British Citizenship. Administrative law is thus concerned, in part, with power and its allocation.

In any application of power, it is to be expected that there will be disagreement over the way in which it is applied. There thus exists the need for a body to adjudicate upon the conflicting claims; claims which may well relate to payments made by the social welfare agencies, the regulation of landlord and tenant relationships, or to provide employment protection. This adjudication could have been left to the existing court system as the body of administrative law built up, but there were a variety of reasons why it was not left to the court system.

First was the view that the court system would not be able to cope with all the disputes, and that the cost of servicing these disputes would be substantial. There was also, as one commentator suggests, "the feeling that the courts might not be altogether sympathetic to the content of some of the legislation, having restrictively interpreted similar legislation in other areas." Whatever the reasons, it was decided that independant (from the government departments, that is) Tribunals were the better alternative to the court system. One such area which has these tribunals is immigration.

Whatever the reasons for the use of tribunals, they had the practical effect of attempting to keep the judiciary out of public policy operation. This restriction on the judiciary was not welcomed by them in the inter-war years, was accepted in the war and immediate post-war years, and is now subject to much judicial hostility as most commentators agree [de Smith, 1980]. The problem for the legislators has been to keep the judiciary out of this public policy operation; the method which they have attempted to use has been the ouster clause. The ouster clause is a piece of text which has been inserted into the legislation with the intent of precluding judicial intervention. However, it has been noted that these formulae have had little effect - they have been ignored or over-ruled by the judiciary.

Now, we have been arguing that the legal process is principally a process of social negotiation; this negotiation occurs, with regard to ouster clauses, in a running battle between the legislators and the judiciary to formalize a clause which will have the desired effect. Thus, not only the meaning of the words which are contained within the clause, but also the practical effect which they might have has been taken into account. Thus, the legislators have understood that the legal text is not sufficient, but that the expected countering strategies of the judiciary must be taken into account - a perfect example of negotiation, we would argue. As examples of some of these

clauses we can look to those which have attempted to exclude the judiciary completely; others have been tried which attempt to limit rather than exclude judicial intervention completely. We use the classification from [Craig, 1983] and his analysis of the court's response.

**Finality Clauses:** these are clauses which attempt to render the effect of a tribunal as "final" or unassailable. The courts gave these short shrift by a variety of methods - for example, Denning L.J. concluded that they only affected appeal, not judicial review. Even this limited effect of the clauses is now viewed as out of date.

**"No Certiorari" Clauses:** Craig [Craig, p518, 1983] writes of these clauses: "Part of the reason for legislative dislike of judicial review was that the courts would overturn decisions for reasons redolent of a Dickensian caricature. Technical error was seized upon and verdicts quashed with an excess of vigour that bordered upon the pedantic. The legislature responded in a number of ways, one of which was the insertion of no certiorari clauses within statutes. Judicial response to such terms was not wholly aggressive: the courts acknowledged that they had been overtechnical. Jurisdictional defects continued to remain unaffected by no certiorari clauses; the courts still struck them down."

**"Shall not be Questioned" Clauses:** this type of clause means what it says, and was often used in conjunction with no certiorari clauses. Of them, Craig [Craig, p520, 1983] writes: "Any hope that persevering parliamentary draftsmen might have had that this formula would work where all else had failed was to prove unfounded."

"As if Enacted" and "Conclusive Evidence" Clauses: these were relatively crafty means of putting a statutory order or a minister's decision on the same level as a piece of legislation and thus, since the judiciary cannot overrule legislation, immune from review. It had some success in the early part of this century but has passed out of fashion. Craig anyhow considers that it is unlikely that it would be successful if brought back into usage - "in the light of the interpretation placed upon other attempts to exclude review, one would hesitate to put too much reliance on them" [Craig, p521, 1983].

So, we argue that there is much in law which is not part of the legislation; the legislation, although clearly stated, does not seem to have much effect upon judicial review of administrative and tribunal practice. But the Prolog team think otherwise. They believe that a useful predictive expert system can be built by simply interpreting the legislation (and, according to [Cory, 1984] having this done by a computer science student rather than a lawyer, it should be noted) and inputting this into a logic program. We argue that this is naive, for the British Nationality Act (1981) too has an ouster clause, and as we have seen above, ouster clauses seem to have no effect. (Not only that, but the section containing the clause is self contradictory - in logical terms - and would provide a substantial problem to logic programmers.) We argue, then, that even if the Prolog team did manage to fully formalize the Act, it would have little predictive power in that part, at least. This, we argue, is a strong rebuttal of the ability to axiomatize an Act of law.

The ouster clause in the British Nationality Act (1981) is in Section 44 (2); the logical contradiction between 44(2) and 44(3):

44 (2) The Secretary of State, a Governor or a Lieutenant-Governor, as the case may be, shall not be required to assign any reason for the grant or refusal of any application under this Act the decision on which is at his discretion; and the decision of the Secretary of State or a Governor or Lieutenant-Governor on any such application

shall not be subject to appeal to, or review in, any court.

44(3) Nothing in this section affects the jurisdiction of any court to entertain proceedings of any description concerning the rights of any person under any provision of this Act.

We suggest that only by recourse to an understanding of the debate over ouster clauses can the logical contradiction between the two pieces of text be understood. That will always be outside the ken of any expert system which contains only axiomatized legislation.

The courts have, for the record, already (given the time scale) carried out judicial review on several cases; this review has been done in the following reported cases. It is important to note that if the ouster clause operated as it "logically" should; these cases should not have been considered for judicial review purposes.

R v Secretary of State for the Home Department ex parte Yates, Queen's Bench Division (Crown Office List), 11 March 1985 (reported on LEXIS). [In this case the difficulty of isolating the 1981 Act from the 1971 Act is shown - thus an axiomatization of the 1981 Act, if such a thing were possible, could not be formally or logically understood without a similar axiomatization of the 1971 Act, which would require further axiomatization ... etc. etc.]

The Queen v The Immigration Appeal Tribunal ex parte Maheswary Chelliah, Queen's Bench Division (Crown Office List), 21 February 1985, (reported on LEXIS). [In this case, the judge termed the argument proposed by the Secretary of State as being an "over legalistic construction" - the judge preferred the common sense construction, it seems, rather than a technically (perhaps logically) correct construction].

Smita Kiritkumar Brahmhatt v Chief Immigration Officer Heathrow Airport Terminal 3, Court of Appeal (Civil Division), 12th December 1984, The Times. [In this case the judiciary involved themselves in considering the proper construction of sections of the 1981 Act - obviously the ouster clause has not worked.]

R v Secretary of State for the Home Department ex parte Parveen, Queen's Bench Division (Crown Office List), 23 October 1984 (report by LEXIS). [In this case the judge took the view that he could not enter into discussion of the meaning of the 1981 Act: "In any event, it seems to me that the matters urged, namely, that rule 47 is somehow inconsistent with the British Nationality Act or is an infringement of the statement of human rights, are not matters in which I am prepared to enter." Forbes J gave as the reason why he felt unprepared to enter into these discussions was that the Immigration Rules were clear about the case in hand - notice that he did not state that it was because of the ouster clause in the Act.]

R v Secretary of State for the Home Department ex parte Syed, Queen's Bench Division (Crown Office List), 25 May 1984 (report by LEXIS). [In this case Mann J. stated that, "I fully accept that the Home Secretary is entitled to have rules of policy which approach or equal the rigidity of a rule. What, however, he is not entitled to do is simply to decline to consider an application from a person who satisfies the pre-conditions imposed by parliament" - this is, of course, a clear attack upon the discretion awarded to the Home Secretary by the legislation].

Of course, this does not mean that the reviews have been successful for those appealing: as Griffith [Griffith, 1985] has pointed out, the courts have not been too keen to side with those from, for example, the Indian sub-continent. But that does not remove the fact the courts have ignored section 44(2) of the Act, and that this ignoring of that section contradicts the claims of legal logic programmers.

As a short aside, we should note that the political character of the judicial process has been noted by such a conservative character as Lord Hailsham. Speaking in a House of Lords debate, he put the position quite succinctly as:

"Judges cannot choose the work they do; they have to come to a decision one way or another on all litigation which is brought before them. If they assume jurisdiction they are in politics; if they decline jurisdiction they are in politics."

[House of Lords, 20th November 1978, Col. 1384]

So, my argument - noted in the next section - about the importance of seeing the judicial role as intrinsically political (eg with their role in judicial review) cannot be argued against as "radical".

We suggest that the approach taken by the ELI program in utilizing the abilities of an expert to reformulate the meaning and effect of an Act achieves an end much closer to that of real law than that promised by logic programmers. Weak behaviourism is better than strong formalization.

The British Nationality Act (1981) has only recently come into force and so there is only a limited understanding of how the judiciary will react to it; we might expect other differences between the legislation and the interpretation of the legislation to appear. In the next section, we argue that even if the legislation was clear and the judiciary did follow it precisely (if such a precision could exist) there are still problems in trying to handle law as a set of clear rules - principally problems of morality and responsibility in a democratic society.

#### 9.6 CONSERVATIVELY APPLYING THE RULES: LOGIC AGAIN

One of the most forcefully presented arguments of the positivist legal philosopher has been that of the relationship of morality to law. They have presented the argument that the technical application of the rules of law is the way in which justice is best achieved.

This concern with justice should, we hold, be a matter of concern to all who are involved in the legal process; even those of us in computer science or logic who have an interest in the law should be concerned with its moral application since many of us entered the research field with a strong inclination to provide tools which increase access to justice, not simply to raise the (not insubstantial) earning power of the legal practitioner.

The question, though, which is raised by positivism is: is technical application of the rules of law the best way to achieve justice? Or might there be some other process which we can concentrate upon which has a "higher moral worth". MacCormick is certainly one of those who is most adamant that justice is best found through logic - using logical reasoning, proceeding step by step through the legal norms. The belief in the utility of this technical process is certainly very strong: recently, even socialists have claimed that the correct way to control the creativity of the judiciary is to ensure that they apply the rules

in a technical manner [Robson, 1981].

This is a view we reject; primarily because it frees the judiciary from the responsibility for any of the judgements which they arrive at - they can say, as they do, "I would like to award judgement to X, but unfortunately I cannot". Judges in our view should be held responsible for their adjudications. They should apply the law, not in a formal manner excluding the social factors and context, but as the legislators and society would wish. We should remember that the War Crimes trials after the second world war were notable for the number of defences which claimed, "I was only applying the rules". Such a defence was not accepted then; and it should not be acceptable now.

The problem is particularly important, we believe, to computer scientists entering the field, because it is a well known fact that "scienfication" and legitimation can often be achieved by the presenting of contentious information in the form of a computer program. We can point to the study by Bloomfield [Bloomfield, forthcoming] which demonstrates how the computer-based Urban Dynamics Model prepared by a team under J.W. Forrester (the inventor of the magnetic memory core) was given extra legitimation in the eyes of administrators. The model proposed that by demolishing low cost housing in American cities, the poor would be removed from the cities thus solving many of the problems of urban deprivation. Of course, because the model only dealt with the city in isolation, it provided little information on what was to happen to these dispossessed poor.

We believe that the presentation of the Prolog project is just as problematical. The language which has been used to present the system is appealing to those who might wish a system which uncritically presents their view as "a logical consequence of the Act". There is to be no escape from the logical compulsion - by interacting with the program people will discover whether they are British citizens or not.



This is, the extreme cynic might suggest, an ideal explication of the governmental position; that extreme cynic might suggest that a government given the opportunity to fund research which presented its law as closed to negotiation or to fund research which was critical of this view would choose the former. But this is not to suggest that the Prolog team are necessarily government funded apologists for right-wing immigration policy. It is, perhaps, more to do with the failure of current jurisprudence to escape the errors of positivism than a conscious decision on the team's behalf to persuade potential users that "the law's the law".

We shall emphasize that "the law's not the law" by looking behind the legislation of the British Nationality Act (1981) to suggest that we cannot simply accept a legal rule and its application without further discussion. Thankfully, there are ways and means whereby a more just solution can be arrived at.

It has been noted by many (non radical) legal commentators on British Immigration legislation that there are implicitly racist overtones involved. For example, one of the stated primary aims of the legislation has been to reduce the incoming numbers of immigrants so that employment prospects for those already in the country are not reduced; yet, if this were so, the commentators have argued, those Irish from Eire should have been subject to the same immigration policies which affected other countries prior to the UK's joining the EEC. In fact they were subject to no restrictions whatsoever. Also, these same commentators have pointed out that even those who are fully qualified for British Citizenship cannot migrate to the U.K. without proper documentation: such documentation can be obtained within a few days in countries such as Australia and New Zealand, but some 12 - 18 months or more in, for example, the Indian sub-continent. The point is that a logical formalization of the British Nationality Act will tell us nothing about these factors.

As well as the legislation having an implicitly racist bias, the British Courts have been keen to support the legislation (which is different from their approach to, say, tax cases). Griffith [Griffith, 1985] has noted the generally limiting view of they have of their interventionist ability in these cases; this limiting view has been upheld in various appeal cases. This can also be seen in those cases which have appeared dealing with judicial review of the British Nationality Act (1981): for example, it could be argued that the current entry clearance required for entry to the U.K. need not necessarily be provided in the country of origin since there is no place in the legislation which says that the potential immigrant must arrive with this documentation. The courts, though, have supported the Home Office policy as implemented by the Immigration Officer and stated that "a current entry clearance was required .. before she entered this country" R v Secretary of State for the Home Office Department ex parte Choudhury. Thus, the courts held that the woman in that case had to return to the country of origin to await proper documentation being prepared, rather than wait for this at Heathrow.

The conjunction of racist legislation and a compliant judiciary might, it could seem, bode ill for any potential immigrant from India or Bangladesh. However, as is currently being shown, there are means to overcome this bias against certain (ie coloured) immigrants. The means comes from EEC legislation; this indicates further that when we wish to analyze the law and how we should react to it, we cannot simply look at one piece of legislation (and formalize that) and expect it to answer all our questions - there are often strategies which the legislation would have us believe do not exist. Logicians, therefore, should not be the first port of call for advice if you have immigration problems, we might suggest.

MacDonald has written of the interplay of EEC and UK law (although, prior to the effects of the 1981 Act):

"The impact on UK immigration law and practice by the EEC has been striking and is by no means fully worked out. British immigration authorities have, on the whole, been brought up and nurtured on a diet of exclusion and restriction. That after all is the history of the alien legislation from the time of the French Revolution, when Jacobins had to be kept out, to modern times when it was French communards, Jews and Germans. It is very much part of the mentality of exclusion embodied in the modern legislation from 1962 onwards. The difficulty for British immigration authorities is to cope with a set of laws, in whose making they played little part, and which is based on an opposite philosophy to their own - on the liberal view that it is right and proper and a good thing for people of different nationalities to move freely from the territory of one member state to that of another. The policy for the European Court had also been a liberal one, extending the scope of the free-movement provision and reducing the restrictive powers of member states. This contrasts with the British courts where the trend of immigration decisions has been to tighten immigration control and extend the power of the state almost to the point of arbitrariness."

[MacDonald, p3, 1983]

And there also exists that other European court which British subjects (including those excluded from the U.K.) have access to - the European Court of Human Rights. Now, whilst this Court has no authority with the legislature, it most certainly does with the Crown who have agreed to be bound by the decisions and practice of that court. The European Court has recently found that the immigration rules breached Article 13 of the Convention on Human Rights because they discriminated against women.

Under the rules, foreign men with full residency rights in the UK could bring in their wives or fiancées, but foreign women cannot. At one hearing the British Government claimed that the purpose of the rules were to protect the domestic labour market at a time of high unemployment. The court ruled that this was a legitimate aim, but provided insufficient grounds for justifying a breach of article 13.

These rules are not directly a part of the 1981 Act, but they do indicate that the word of the legislature is not final. Given the time lag between the beginning of an appeal or judicial review to possible judgement at the European Court in Strasbourg, we might expect the 1981 Act to be soon commented upon by this Human Rights court. The logic programmers with their "logical consequences" can tell us nothing about the likely eventualities of such a process. Hence, we can only reiterate the advice to potential immigrants or those concerned with their nationality to discuss the problem with an expert; not with a legal rule-based expert system. The advice they would get could not possibly be worse.

We would suggest that for all the limitations of the ELI project, it did at least consider some of these problems of law in its design. That we eventually claim that even this extended and annotated type of rule-based system is too weak for real handling of law, should not detract from the fact that we were aware of some of the problems inherent in applying rule-based strategies to the law. The remaining problems came to light only on analysis of the completed system.

## 9.7 SYNOPSIS OF THE CHALLENGE TO LEGAL LOGICIANS

The content of this chapter has been an attack upon the very notions of logic and rules in the law. We have suggested that the problems with these approaches are of a deep and substantial nature - in effect, that tinkering with programs such as the ELI program will not overcome the

limitations. Rather, if we wish to use ELI type systems, we shall have to find a para-legal situation for them (Chapter 8).

But we have also gone further and suggested that logic programming has no place at all in the law - for it is incapable of presenting secondary and contextual information of the sort which a system such as ELI might, with its annotations and secondary information contained within the system. Now these are substantial criticisms of claims which are currently being made by logic programmers; we should apply a surgical technique to this chapter and list the objections which are made: then, these researchers will not be under any illusion about the major criticisms which they must resolve in order to disprove the argument.

The challenge can be presented as:

- 1) Explain how the formalization of legislation relates to the legal process: thus, why is it that there are sections of legislation (such as ouster clauses) which ought to have an effect when read in the usual manner (or in a "logical" manner, even), yet have little of their intended effect in practice.
- 2) Explain how logical contradictions can appear in the body of legislation (such as appears in the British Nationality Act (1981)) and yet the legislation can still be used to offer a consistent and coherent perspective from which the judiciary can adjudge.
- 3) Explain how the legislation has a constraining effect upon the judiciary, when that judiciary will interpret the legislation in the light of common law principles (since, as our view has it, the judiciary are the legal dictionary which gives meaning to the legislation).
- 4) Explain how the notions of morality and justice can be handled by a series of logical sentences. Thus, should judges apply the rules without considering the legal and social context within which those rules were made; or should they, as Lord Denning has proposed, find some way of overcoming the rules to reach a just settlement. If the latter, explain how might this be done by logic?

I would accept these explanations in the form of a logic program which could convincingly be used by legal practitioners, or even by para-legal practitioners. But I do not expect to see such a program, no matter how great the claims which have been made and are being made for logic programming. I do not expect to see this program, because logic programming is flawed by an errant epistemology - it sees the world in

terms of a computational, logical model; and when that model meets the real world it fails because the world is neither a logical nor a computational world.

## CHAPTER 10

### CONCLUSION: AFTER RULE-BASED SYSTEMS?

#### 10.1 RESEARCH GOALS

The goals of research teams in this field differ fundamentally. Some might wish to provide powerful computer systems which can answer the user on any question in law that might be put to it (as was suggested by Mehl). Others might wish to formalize the law into some language which allows conceivable logical conclusions to be tested. Yet another goal might well be a system which attempts to predict individual judicial behaviour. All these goals will require the setting of different questions which must be asked to assess potential success or failure; yet, there will be a core question which can be asked of all those systems which attempt to represent "the law". That question is quite simply,

"is it the law or the legal process which the computer system is actually describing, or is it some kind of limited pseudo- or para-legal model which is being represented?"

I have tried to argue that logical and rule based methods are not appropriate or powerful enough means to represent either the law or the legal process. If my argument is accepted, then how might researchers pose further questions to determine when they have produced powerful legal expert systems which do handle properly legal questions? I would argue that the two questions are:

(1) Is the system capable of generating legal rule breaking strategies - Can counter rules, reasons for ignoring rules, contrary interpretations be provided, for example, by the system?

(2) Is the system capable of generating legal rule supporting strategies - Can supporting interpretations of rules, reasons for supporting rules, examples of contrary interpretations which have been declined in earlier cases be provided, for example, by the system?

since my view of the legal process is that these two aspects of rule breaking and rule supporting are fundamental - it is those two aspects which are at the heart of the law. It is therefore those two aspects which are of fundamental importance to any system which claims to rise above the purely clerical to that level of providing "legal advice". For rules are only a shorthand means of communicating information: rules miss out so much of the necessary contextual information. For proper advice it is necessary to go beyond the shorthand to something more verbose and informative.

Since I now propose that the legal process is a negotiatory process and not one which is principally rule based, I must return to the point made earlier concerning the importance of research questions - in this new field of legal knowledge engineering, it is essential that we do not forget that there were two questions:

(a) what is it about the law that we are trying to represent within our computer programs?

(b) how can our assumptions about what the law is be represented in a computer program?

I started off, in the initial research phase, answering question (a), with the proposition that the law is rule based; this resulted in answering question (b) with the use of rule-based methods. Having a new



answer to question (a), my ideas about the solution to (b) have, of course, to be revised. They are not yet in a completed form. But this is the nature of research - entering a dark cupboard, sure of what you wish to find yet frequently coming out with something different or, rarely, nothing at all (since it is usually possible to learn from mistakes).

Given this, how might we view the next stages in the ELI project - how might the project work towards the provision of a legal expert system?

## 10.2 DETERMINISM AND NON-DETERMINISM

It is useful to use the dichotomy of determinism/non-determinism (of the non-mathematical sort) to discuss possible types of system.

Deterministic systems are those which encapsulate one particular framework of knowledge and present it in a pedagogical manner. Such systems accord with the classic expert system model - the user of such a system is cast in the role of recipient where the expertise of the system is proffered on a "take it or leave it basis". Such deterministic systems are, of course, those which we have been referring to as rule-based in this paper. But such systems cannot be as potentially powerful in legal areas as they might be in non-legal areas (such as medicine, engineering etc.) for the very reason that the legal process is not, I have argued, an area where people have an interest in coming to agreement about the "facts", "rules" or "principles" of a case; rather, the interest which those involved in the case have is one of argument over those facts, rules and principles.

But that does not mean that rule-based systems are without any use at all; it only means that in the legal process proper (i.e. where the barrister presents a case in court) they are of little use. We can imagine that they might have a use in a semi-legal situation, where a given interpretation of the law is handled. Niblett has proposed that

their use as contrasting expert models:

"It is a mistake to suppose that one expert system is sufficient for one area of law. Some systems, like some lawyers, will be better than others for they will have a more refined knowledge base and superior powers of reasoning. Just as some lawyers are plaintiff's men whilst others are more at ease representing defendants, so some expert systems will be tuned to the requirements of plaintiffs and others to defendants. Some tax systems will favour the Revenue whilst others will be more suited to the taxpayer. Of course if two expert systems differ on a point of law, then when that issue has been resolved (for example by judicial decision) then one or both of their knowledge bases must be modified to include the new result."

[Niblett, 1981]

I would argue that, yes, different legal rules and suchlike can be inserted into a legal expert system, but the adding of these rules will do very little to aid the user in court - simply having a rule inserted into a knowledge base cannot help the defendant to present his case. He cannot hold up the printout and cry, "My rule clearly says .... " for it is impossible to tell for how long "that issue has been resolved (for example by judicial decision)". The judge, also, may well decide that that's not what the rule actually is, or that another rule is more important.

We can imagine deterministic legal expert systems having some, if limited use, as a clerical system which presents to, say, employees of the D.H.S.S. or some other governmental service. Then, users might be able to refer to the system as an aid in applying the rules; these rules are, of course, those which the governmental agency would wish to apply - they would still be open to legal challenge through the legal process proper. Typically, in the UK at present, such challenge is less easy

than under previous political masters but it can be done - for example, challenges through European courts frequently point out the utility to the appellant of not accepting "the rules".

Thinking beyond the deterministic system it might be considered how systems might be built which can aid the person trying to prepare a case. The system might do this by helping to pinpoint the various potential outcomes - or rule breaking strategies. If a user interacts with a deterministic system, only one of those outcomes can be proposed (what the expert who trained the system - in a rule based system this might simply be an interpretation of the legislation - would do in that case). Experts are often wrong, so it seems essential to build legal advisory systems which can encapsulate various expert models, and which might allow the user to add to the expertise that he or she has to the decision process - in other words, we are basing our systems upon a "textbook model" rather than a rule based model.

Legal textbooks have a substantial advantage over primarily rule based legal systems: they can offer a substantial amount of contextual information - they can provide an introduction, an index to the parts, and much in the way of anecdotal extracts from case reports. But a legal textbook does not attempt to tell the user how to apply the rules; it might well be that this is the only advantage that rule based expert systems can offer the lawyer.

Taking this textbook analogy further (and we should note that the first stop in legal research is usually a textbook, it is not simply something utilized by undergraduates) we might propose that trying to incorporate that sort of information into a computer system (together with any other useful information) might well be the way in which we can produce non-deterministic "expert" systems.

One of the important aspects of this type of system would be that the suppliant role of system and user is broken down, and the computer attempts to use as much of the human ability for intuition as possible. In the type of systems we propose, the intuition is dependant upon the fact that the users will share the common culture of being legally trained. Such a non-deterministic system can be described as a decision support tool; it operates within a framework, where the law is seen to be a complex social process which cannot be totally or formally modelled. Therefore, the user's non-formality ("intuition") should be seen as an intrinsic part of the system.

There should be some extra advantages accruing from the use of computer tools though, even if only to make the expense worthwhile. What might they be? For one thing, the systems might make use of and interface with legal data bases. Commentaries and suggested interpretations of the law might be handled (and easily updated). Also, other factors can be handled which cannot be held in a textbook; for instance, the builder of the system may inform the program that an extremely conservative bias might well be present in the eventual judicial decision, or that past experience has shown Judge X to have a view Y on the type of problem under investigation (see for example [Lawlor, 1980] and [Nagel, 1977] for mathematical techniques which have been used with some success in this area).

Such non-deterministic systems are perhaps more complex to design (and for experts to pass their expertise to) than rule based deterministic systems, but we see this type of system as that which will best accord with the needs of the lawyers.

Of course, such an outline admits that we are no longer attempting to design systems with "artificial intelligence"; but that may be no such bad thing.

There is a practical problem in attempting to discover the needs of the legal profession relating to how they actually do research and what the materials of research are. Some limited work has been done; for example [Foster, 1984] and some (sadly as yet) unpublished research by Colin Campbell (Queen's, Belfast). Though a large number of texts have been written covering access to legal research materials, and texts have been produced describing legal research for students, they offer little insight into actual legal research.

There seems to be some suggestion from Campbell's research that little legal research is carried out by solicitors (this is of course relevant in the English, Welsh and Northern Irish context); whenever a real problem of law appears an outline of the situation is passed onto counsel for opinion. This puts the solicitor in the position of the general practitioner from medicine; whenever a medical/legal problem is met this is passed onto counsel/consultant.

If we are to look for alternatives to rule based methods in legal knowledge engineering, I would argue, it is vital to extend the quantity of empirical study into just what the real needs of the profession are. I would be unhappy to learn from this research that there was indeed very little need for computer tools; but at least then we would not be following a wild goose; a wild goose that told us it knew which rules of law were clear.

## APPENDIX A

### THE ELI COMMANDS

#### A.1 INTRODUCTION

The specific functions of the system are detailed below. Generally, the program has been designed to be easy to interact with - something which was made relatively easy by use of the INTERLISP language. INTERLISP can be seen as a large collection of programmer's utilities surrounding a LISP core [Sandewall, 1976], each of the utilities being accessible both to aid the programming and debugging process and also available from within LISP programs themselves. In several parts of the program these utilities have been used - the structure editor is used to edit rules for example. Another example is the use of the ASKUSER facility which allows correct input to be explicitly stated; any other input is therefore ignored.

Within the program, any operation can be cancelled and the user left at the next higher level simply by giving the "OK" command. At any point in the interaction, the user can receive a "help message" by typing "?". The help message contains information about the valid inputs and options.

The system does not currently incorporate an UNDO facility but, should this be required, it can be relatively easily incorporated by using the history lists and UNDO facility which INTERLISP possesses. As a testbed system, however, such a requirement was not considered essential; it

would be necessary in any usefully implemented legal consultative system.

We here describe, for each of the program's modules, the most important commands which are available to the user. In the examples given, user input is underlined, although usually only the first one or two letters are required before unambiguous recognition is achieved.

## A.2 THE SETUP MODULE

The SETUP module is used by the expert to input/edit the rules prior to their being input to the knowledge base.

```
setup]  
Current, old or new file: new file to be set up  
Name of new rules file: test.rule  
The last rule entered has the rule number 0  
There are 0 rules which have not been entered to the rules file.  
SS:
```

1) WRITE - prepare and name a rule ready for input or editing.

Operation:

When first called, the system prompts with the number of the new rule - an increment on the number of the last input rule.

For each condition to be input the system prompts "Condition:" and requires the input text to be enclosed within parenthesis - e.g. "(this is a condition)". To terminate input of the rule and discard any conditions which have been input, the subcommand "OK" is given; to terminate condition input (and then input the goal), a null list must be input.

The system then prompts "Goal:" for the goal. The goal input is similar to that of the condition (i.e. it must be parenthesized and the subcommands "OK" and "()" operate in the same manner).

When properly input, the rule name is printed on the terminal (e.g. "RULE-1").

Example:

```
SS: write rules.
    This rule will be numbered 1
    Condition: (this is the first condition)
    Condition: (and this is the second)
    Condition: ]
    Goal: (this is the goal)
    Rule is named: RULE-1
    This rule will be numbered 2
    Condition: ok
SS:
```

2) EDIT - edit a rule which has been input. This command makes use of the INTERLISP EDITV facility (a structure as opposed to text editor) and uses INTERLISP EDIT commands. The structure editor's prompt is "\*".

Operation:

To edit a rule the rule number should be input. The system will check that the rule actually exists before entering the editor.

While the INTERLISP EDIT facility is large and powerful, it is possible to achieve simple, positive editing with a few subcommands:

The rule is held as the following LISP S-expression:

```
(IF ((c1)...(cn)) THEN (goal))
```

where (c1) to (cn) are any number of conditions. Moving about this structure requires only the following commands:

To go to the first element in the rule (i.e. "IF") input "1"; for the second (i.e. the list of conditions) input "2", etc. To go to the "top level" input "" (the up arrow).



When accessing the conditions, these numbers can be used in the same manner - to go to the first condition, input "1" etc.

These commands only move about the structure. To print out any of the items in the structure input "p" or "pp" (for prettyprint).

Deleting an element in the structure is done with a colon (":"). Anything appearing after the colon on the EDITV input line is used to replace the deleted structure.

To insert something in the structure, use "A" (meaning, insert After this element) or "B" (inset Before this structure) followed by the item(s) to be input.

For other commands see the INTERLISP manual.

Example:

```
SS: edit rule
    Rule number for rule to be edited: 1
edit
1*pp
(IF ((THIS IS THE FIRST CONDITION)
    (AND THIS IS THE SECOND))
    THEN (THIS IS THE GOAL))
1*2 1
2*p
(THIS IS THE FIRST CONDITION)
2*3 a very
3*p
.. THE VERY FIRST CONDITION)
3*ok
SS:
```

3) DELETE - rules which have been input are held on the rule list - this command removes them from this list. When the rules file is next written the deleted rule is not written to it.

Operation:

The system prompts for the number of the rule which is to be deleted. If it exists (has been input and not already deleted) then it is printed out, then deleted.

4) COPY - makes a copy (with, of course, a new name) of a rule already written. In combination with the EDIT command this allows rules with slightly different elements to be quickly input.

Operation:

The system prompts for the number of the rule which is to be copied. On receipt of this, a copy is made and the copy is named.

5) DIRECTORY - lists the filenames held in the user's directory. This will print out all filenames - not only those used by the system. In other parts of the system when the DIRECTORY command is given only those .RULE, .KNOW or .LSP files in the directory are named.

Example:

```
SS: directory of files
    <RL.LEITH>
ANIMALS.LSP.1
BUILD.LSP.2
DESIGN.RNO.1
EEC.DIC.2
HEALTH.RULE.2
HOUSEBEN.RULE.2
INTERPRET.LSP.7
SETUP.LSP.3
SOCIAL.RNO.7
SS:
```

6) UPDATE - writes to the file which will contain the rules. When the SETUP function is called it will check to see whether a suitable rules file has been loaded. If there is no suitable file, then the system will set up a new one, or load an old file which already contains some rules. Thus several rules files can be kept in the directory and manipulated easily.

Operation:

The system informs the user that it is updating a named file.

Example:

```
SS: update rules file  
    Writing to TEST.RULE  
SS:
```

7) INITIALIZE - removes and initializes all rules from rule list after preparing a named backup file.

Operation:

The system first prompts for the name of a backup file to which it can write the present rules. This file is then written to, and all relevant lists within working memory are initialized.

Example:

```
SS: initialize rules file  
    Name of backupfile: backup.rule  
    Current rule collection now being written to BACKUP.RULE  
    Writing to TEST.RULE  
SS:
```

8) PRINT - prints any specified rules or all rules. If all rules are to be printed, then they can either be written to the terminal or to a named file.

Operation:

The system prompts for either the number of the rule to be printed or the name of the output file (which is "T" if the terminal is to be used). If a number is input then that rule is printed on the terminal (if it exists). If the input is neither a number nor "T" then the system will prettyprint the rules to that text file. The file must exist before the rules can be printed to it. The rules are printed in the order in which they appear in the rule list.

Example:

```
SS: print rules
    Print request: t
RULE-1  (IF ((THIS IS THE VERY FIRST CONDITION)
            (AND THIS IS THE SECOND))
        THEN
        (THIS IS THE GOAL))
RULE-2  (IF ((THIS IS THE FIRST CONDITION OF THE SECOND RULE)
            (AND THIS IS THE SECOND OF THE SECOND RULE))
        THEN
        (AND THIS IS THE SECOND GOAL))
SS:
```

9) MOVE - when the rules held on the rule list are input to a knowledge base they are input in the order in which they appear on rule list. This facility enables them to be moved about the list easily.

Operation:

The system prompts for the number of the rule which is to be moved; then whether it is to be moved to "before" or "after"; and then the rule number which is to be moved in relation to. An error is reported if either of the rules do not exist.

Example:

```
SS: move rules in rule list
    Move which rule: 1
    B(efore) or A(fter): a
    Which rule: 2
SS: print rules
    Print request: t
RULE-2  (IF ((THIS IS THE FIRST CONDITION OF THE SECOND RULE)
            (AND THIS IS THE SECOND OF THE SECOND RULE))
        THEN
        (AND THIS IS THE SECOND GOAL))
RULE-1  (IF ((THIS IS THE VERY FIRST CONDITION)
            (AND THIS IS THE SECOND))
        THEN
        (THIS IS THE GOAL))
SS:
```

10) SEARCH - when the number of rules in rule list grows, it is useful to be able to search for instances of various words and phrases. The SEARCH facility allows this, printing out either one or more rule numbers or rules themselves.

## Operation:

The system prompts for the search strategy - the first incidence of a string, one specific incidence or all incidences. If all incidences are required, the user is then asked whether only rule numbers are to be printed on the terminal, or whether the rule texts are to be printed on the terminal or a file.

Having this information the user then specifies the string to be searched for (specifying it between inverted commas). Searching then begins from the first rule in rule list onwards.

Depending upon which search strategy was chosen, the output information is presented to the user.

### Example:


```
SS: search for text in rules  
   Search request: first incidence  
   String to be searched for: "first"  
   String found in rule RULE-2:  
(IF ((THIS IS THE FIRST CONDITION OF THE SECOND RULE)  
      (AND THIS IS THE SECOND OF THE SECOND RULE))  
      THEN  
      (AND THIS IS THE SECOND GOAL))  
SS:
```

## A.3 THE BUILD MODULE

The function of the BUILD module is to allow the expert to input rules to the knowledge base and to "fine-tune" the advice produced from the knowledge base.

When the BUILD facility is called it requests information from the user on which knowledge base is to be used or, if a new knowledge base is required, what the file to hold it should be called. having this information, the system can then indicate how many rules are currently in the knowledge base:

```
build]  
Current knowledge base or New or Old file? new file to hold new
```

knowledge base  
Name of file:: test.know  
Delete any word and concept lists? yes  
Highest rule-number in KB is: 0  
BB: 

When in the BUILD environment (where the prompt is "BB:") the following commands are available.

1) INSERT - insert rules into the knowledge base structure. These rules can either be brought from a rules file, written and edited or be individual rules which already exist.

Operation:

The user is prompted for information on which format the rules are to be input - either from a file prepared by the SETUP function, as a named rule, or made up. The latter two are useful when fine-tuning the system. The former is most useful when in the initial stages of building the knowledge base.

Rules can be input from a file which has previously been used as input - information on which rules have been input from that file is kept in the rule file; if this is the case, the user is asked whether all the rules or only those which have not been input should be input.

The user is asked whether words should be added to the words list prior to input from the file starting, or whether the symbols should be extracted as the individual rules are being input.

As the rules are being input, the number of the rule is printed together with whether the rule was input top down ("T"), goal up ("G") or alone ("I").

Example:

BB: insert rules into knowledge base  
Input request: from file  
Name file containing rules: animals.rule  
FILE CREATED 18-Sep-83 14:02:12

#### RULESCOMS

18 rules in this file have not yet been input

Rule numbering starts at: 1

After how many rules have been input do you wish the knowledge  
base file and rule file to be updated: 10

Extract words before inputting rules: Y/N: yes

.....I1I2I3I4I5I6T7T8T9T10

Knowledge base being updated to: TEST.KNOW

Writing to ANIMALS.RULE

I11T12I13T14T15I16I17I18

Knowledge base being updated to: TEST.KNOW

Writing to ANIMALS.RULE

Done.

BB:

2) DELETE - deletes rules from the knowledge base.

#### Operation

The rule number to be deleted is prompted for. If it exists, then it is removed from the knowledge base. An option which is available allows the user to search for rules containing a string before deleting them.

#### Example:

BB: delete rules from knowledge base

Deletion request: Rule

Delete rule: 1

BB: print rules from knowledge base

Print request: 1

Sorry, no rule number 1

BB:

3) INTERROGATE - this calls the INTERPRET module and is useful for moving between knowledge base editing and fine-tuning, allowing the user to oscillate between testing and amending.

The operation of the INTERPRET module is dealt with in the next subsection.

4) DIRECTORY - this prints out all the file names of files held in the user's directory. The operation of this command is identical to that in the SETUP module.

5) PRINT - prints rules from the knowledge base to the terminal or to a file.

Operation:

The operation of the print facility is similar to that in the SETUP module, differing in only the respect that when printing out the rules from the knowledge base, the rules can either be printed in rule number order, or the order that they are stored in the knowledge base.

Example:

```
BB: print rules from knowledge base
    Print request: 4
    (IF ((ANIMAL EATS MEAT))
      THEN
        (ANIMAL IS CARNIVORE))
BB:
```

6) INITIALIZE - removes all rules from the knowledge base structure after writing to a backup file.

Operation:

The name of the backup file is first prompted for, then every relevant list and data structure is initialized.

Example:

```
BB: initialize knowledge base
    Confirm to delete all rules: Y/N: yes
    Name of backup file: backup.know
    Writing backup to backup.know
    Deleting rules from knowledge base.
    Delete all word and concept matching lists: Y/N: yes
BB:
```

7) NOTATE - is used to either write annotations to a rule or rule element or to read one of the already written annotations.



Since much manipulation of the knowledge base will be done during the fine tuning of the system, it is inadvisable to add notations until the system is fairly well organised and defined.

Operation:

Writing annotations:

When the notate command is given the user should give the "Rule" or "Condition" subcommand which, respectively, allow either an entire rule to be annotated or an element (including the goal). For both options the rule number must be specified, and to annotate an individual element of the rule the element must be specified. When this is done input from the terminal is requested; this can be in either upper or lower case.

Example:

```
BB: notate rules and rule parts
Annotation request: condition or goal to be annotated
Rule-number: 1
  1 (ANIMAL GIVES MILK)
    Current annotation:
  2 (ANIMAL IS MAMMAL)
    Current annotation:
Which to be annotated: 1
Input text; paragraph with ''. Terminate by ZZZ:
we do not necessarily mean cows milk here
ZZZ
Rule-number: 1
  1 (ANIMAL GIVES MILK)
    Current annotation:
    we do not necessarily mean cows milk here
  2 (ANIMAL IS MAMMAL)
    Current annotation:
Which to be annotated: 2
Input text; paragraph with ''. Terminate by ZZZ:
mammals are those animals whose young are not
born from eggs ZZZ
BB:
```

Printing annotations:

To print all the annotations connected with a rule (that is both the annotation to the rule and to the elements of that rule) the "Print" subcommand is given.

Example:

BB: notate rules and rule parts

Annotation request: print annotations

Rule number: 1

Rule annotation:

Other annotations are:

1 (ANIMAL GIVES MILK)

we do not necessarily mean cows milk here

2 (ANIMAL IS MAMMAL)

mammals are those animals whose young are not born from eggs

BB:

Editing annotations:

To edit an annotation the rule number must be given just as when printing and inputting the notations. Entering the notation editor is then done with the "Edit" subcommand, followed by either "Rule" or "Condition" for, respectively, to edit a rule or rule element annotation.

The annotation editor uses, like the editor in SETUP, the INTERLISP EDITV facility. These editing commands are described above.

8) ADD WORDS - is used to add words to the "important", "noise" or "permanent noise" word lists. These lists are used by the conceptual matching module within the system. This facility is useful during the BUILD process, where the expert may realize that the prepared rules do not specify a word as either "important" or "noise". Using this command is the only means of adding words to the permanent noise list.

In addition to allowing words to be input to the lists, this facility will also allow these lists to be printed out or for specific words to be deleted from the lists.

Operation:

To add words, the subcommands "NOISE", "IMPORTANT", "PERMANENT" allow words to be added to the relevant lists.

Example:

BB: add to or delete words from word lists  
Word request: noise word  
Word: this  
BB:

To print word lists, the "PRINT" subcommand should be given.

To delete a word from one of the lists give then "Delete" subcommand and then specify which of the lists the word is to be removed from, and then the word itself.

#### A.4 THE INTERPRET MODULE

The INTERPRET module uses the knowledge base which has been created by the BUILD module and "interrogates" it, i.e. it allows advice to be given from the knowledge base.

Like the other two modules the INTERPRET module requests information on whether a new file is to be loaded, or the currently loaded file (if any) is to be used:

When in the INTERPRET environment, the following commands are available. The "?" and "ok" commands operate in this environment as they do in the two earlier ones.

- 1) INITIALIZE - this sets all relevant interrogating information to initial values, and puts traversal information into the knowledge base.
- 2) PRINT - print rules. This operates as the PRINT command in the BUILD module.
- 3) COMMENT - allows the user to determine the amount of information about processing to be printed. "All" information is obviously required by an expert fine-tuning the system, "SOME" when only some is required and "NONE" when only information is required. To find out the current state of the COMMENT flag, give the "?" command.

4) MATCHING - switch between full and reduced conceptual matching.

This switch allows the user to determine whether the conceptual matching module of the system should be used. In the initial stages of testing it provides a useful means of matching conditions, but not in later stages.

5) EDIT - edits concept matching lists, as in the BUILD module.

6) START - begins interrogation of the knowledge base. This command must be used after the knowledge base has been initialized; although, one interrogation can be halted (by the OK command) and restarted at the point which it was halted so long as the system is not reinitialized.

7) INFORMATION - on the various goals/conditions proven true or false and other aspects of the interpretation can be got at the Interpret level or whilst the system is waiting for a decision on whether a presented condition is true or false. The amount of information available is comprehensive; the "?" command will list all relevant information which can be output.

APPENDIX B  
A FRAGMENT OF ELI IN USE

In this section we give an example of a small section of the ELI rule corpus being printed out from the SETUP module, then input into a knowledge base held in the file TEST.KNOW. After this, the knowledge base is interpreted. We have chosen this small section of the entire rule corpus because it represents one small coherent section of that corpus (dealing with heating additions), and because it represents well the use of the negation handling. In order to demonstrate this latter operation, full information on interpretation is printed out - a situation which mars clarity; for an example without this detail being printed, see the following Appendix C.

```
Setup, build or interpret: setup
Current, old or new file: old file to be loaded
Name of old rules file: Directory
Your .RULE files are:
  <RL.P-LEITH>
  ADDREQU.RULE.4
  CERTHOUSEBEN.RULE.3
  FUEL.RULE.12
  HEALTH.RULE.2
  HOUSEBEN.RULE.1
  ODDDEF.RULE.2
  REQUCALC.RULE.6
  RESOURC.RULE.3
  WELFARE.RULE.8
  Current, old or new file: old file to be loaded
  Name of old rules file: FUEL.RULE
  FILE CREATED 27-Nov-84 12:50:17
  RULESCOMS
  The last rule entered has the rule number 11
  There are 0 rules which have not been entered to the rules file.
  There are 11 in this file which have been inserted
  to the knowledge base held in the file OLD.KNOW
  Do you wish to re-enter these rules? yes
  SS: print rules
```

Print request: T

RULE-1 (IF ((APPLICANT IS QUALIFIED AS CLAIMANT)  
 (NIL APPLICANT PAYS FOR HEATING IN RENT)  
 (APPLICANT'S HOME IS HEATED BY CENTRAL HEATING SYSTEM)  
 (NIL APPLICANT'S HOME HAS FIVE OR MORE ROOMS -  
 EXCLUDING BATHROOM, LAVATORY AND HALL)  
 (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 8.2)  
 (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 5.05)  
 (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 4.1))  
 THEN  
 (APPLICANT IS ENTITLED TO HEATING ADDITION OF 2.05))

RULE-2 (IF ((APPLICANT IS QUALIFIED AS CLAIMANT)  
 (NIL APPLICANT PAYS FOR HEATING IN RENT)  
 (WARMTH IS REQUIRED DUE TO ILL-HEALTH OF APPLICANT OR  
 DEPENDANT - OR MOBILITY IS RESTRICTED DUE TO  
 SOME PHYSICAL REASON)  
 (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 8.2)  
 (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 5.05)  
 (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 4.1))  
 THEN  
 (APPLICANT IS ENTITLED TO HEATING ADDITION OF 2.05))

RULE-3 (IF ((APPLICANT IS QUALIFIED AS CLAIMANT)  
 (NIL APPLICANT PAYS FOR HEATING IN RENT)  
 (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 8.2)  
 (WARMTH IS REQUIRED DUE TO SERIOUS ILLNESS OF APPLICANT  
 OR DEPENDANT - OR APPLICANT OR DEPENDANT IS  
 HOUSEBOUND))  
 THEN  
 (APPLICANT IS ENTITLED TO HEATING ADDITION OF 5.05))

RULE-4 (IF ((APPLICANT IS QUALIFIED AS CLAIMANT)  
 (NIL APPLICANT PAYS FOR HEATING IN RENT)  
 (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 8.2)  
 (APPLICANT'S HOME IS DIFFICULT TO HEAT ADEQUATELY,  
 PARTICULARLY DUE TO ROOMS BEING DRAUGHTY,  
 DAMP OR VERY LARGE)  
 (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 5.05)  
 (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 4.1))  
 THEN  
 (APPLICANT IS ENTITLED TO HEATING ADDITION OF 2.05))

RULE-5 (IF ((APPLICANT IS QUALIFIED AS CLAIMANT)  
 (NIL APPLICANT PAYS FOR HEATING IN RENT)  
 (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 8.2)  
 (APPLICANT'S HOME IS EXCEPTIONALLY DIFFICULT TO HEAT,  
 PARTICULARLY DUE TO IT BEING VERY DAMP OR  
 IN A VERY EXPOSED POSITION))  
 THEN  
 (APPLICANT IS ENTITLED TO HEATING ADDITION OF 5.05))

RULE-6 (IF ((APPLICANT IS QUALIFIED AS CLAIMANT)  
 (NIL APPLICANT PAYS FOR HEATING IN RENT)  
 (APPLICANT'S HOME IS HEATED BY CENTRAL HEATING SYSTEM)  
 (APPLICANT'S HOME HAS FIVE OR MORE ROOMS - EXCLUDING  
 BATHROOM, LAVATORY AND HALL)  
 (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 8.2)  
 (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 5.05))  
 THEN  
 (APPLICANT IS ENTITLED TO HEATING ADDITION OF 4.1))

RULE-7 (IF ((APPLICANT IS QUALIFIED AS CLAIMANT)  
 (NIL APPLICANT PAYS FOR HEATING IN RENT)  
 (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 8.2)  
 (APPLICANT'S HOME IS ON AN ESTATE CONSIDERED BY DHSS TO  
 HAVE HIGH HEATING COSTS)  
 (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 5.05)  
 (NIL APPLICANT'S HOME HAS FIVE OR MORE ROOMS - EXCLUDING

BATHROOM, LAVATORY AND HALL))

THEN  
 (APPLICANT IS ENTITLED TO HEATING ADDITION OF 4.1))

RULE-8 (IF ((APPLICANT IS QUALIFIED AS CLAIMANT)  
 (NIL APPLICANT PAYS FOR HEATING IN RENT)  
 (APPLICANT'S HOME IS ON AN ESTATE CONSIDERED BY DHSS TO  
 HAVE HIGH HEATING COSTS)  
 (APPLICANT'S HOME HAS FIVE OR MORE ROOMS - EXCLUDING  
 BATHROOM, LAVATORY AND HALL))

THEN  
 (APPLICANT IS ENTITLED TO HEATING ADDITION OF 8.2))

RULE-9 (IF ((APPLICANT IS QUALIFIED AS CLAIMANT)  
 (APPLICANT OR DEPENDANTS IS RECEIVING MOBILITY OR  
 ATTENDANCE ALLOWANCE OR EQUIVALENT PAID WITH  
 A SERVICES OR DISABLEMENT PENSION)  
 (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 8.2)  
 (NIL APPLICANT PAYS FOR HEATING IN RENT))

THEN  
 (APPLICANT IS ENTITLED TO HEATING ADDITION OF @ 5.05 PER  
 PERSON IN HOUSEHOLD RECEIVING MOBILITY OR  
 ATTENDANCE ALLOWANCE OR SERVICES OR  
 DISABLEMENT PENSION))

RULE-10 (IF ((APPLICANT IS QUALIFIED AS CLAIMANT)  
 (APPLICANT'S HOUSEHOLD CONTAINS SOMEONE OVER 70 OR UNDER  
 5 YEARS OLD)  
 (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 5.05)  
 (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 8.2)  
 (NIL APPLICANT PAYS FOR HEATING IN RENT))

THEN  
 (APPLICANT IS ENTITLED TO HEATING ADDITION OF 2.05))

RULE-11 (IF ((APPLICANT IS QUALIFIED AS CLAIMANT)  
 (APPLICANT PAYS A SEPARATE FIXED CHARGE FOR HEATING,  
 WHICH IS NOT SUBJECT TO REBATE OR SURCHARGE))

THEN  
 (APPLICANT PAYS HEATING IN RENT))

SS: ok

Setup, build or interpret: build

Current knowledge base or New or Old file? new file to hold new  
knowledge base

Name of file:: TEST.KNOW

Delete any word and concept lists? no

Lists remain

Highest rule-number in KB is: 0

BB: insert rules into knowledge base

Input request: from file

Current file is FUEL.RULE

Use current rule file: Y/N: yes

Rule numbering starts at: 1

After how many rules have been input do you wish the  
 knowledge base file and rule file to be updated: 20

Extract words before inputting rules: Y/N: yes

.....I1T2T3T4T5T6T7T8T9T10T11

Knowledge base being updated to: TEST.KNOW

Writing to FUEL.RULE

Done.

BB: ok

Setup, build or interpret: interrogate

Current or new knowledge base: current knowledge base to be  
used

File name is: TEST.KNOW

CC: initialize

CC: start interpretation

Current condition under consideration is

(APPLICANT IS QUALIFIED AS CLAIMANT)

Is this true?::

APPLICANT IS QUALIFIED AS CLAIMANT

:: yes

Current condition under consideration is

(NIL APPLICANT PAYS FOR HEATING IN RENT)

Is this true?::

APPLICANT PAYS FOR HEATING IN RENT

:: no

Current condition under consideration is

(APPLICANT'S HOME IS HEATED BY CENTRAL HEATING SYSTEM)

Is this true?::

APPLICANT'S HOME IS HEATED BY CENTRAL HEATING SYSTEM

:: no

Current condition under consideration is

(WARMTH IS REQUIRED DUE TO ILL-HEALTH OF APPLICANT OR  
DEPENDANT - OR MOBILITY IS RESTRICTED DUE TO  
SOME PHYSICAL REASON)

Is this true?::

WARMTH IS REQUIRED DUE TO ILL-HEALTH OF APPLICANT OR DEPENDANT  
- OR MOBILITY IS RESTRICTED DUE TO SOME PHYSICAL REASON

:: yes

Current condition under consideration is

(NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 8.2)

Trying to prove the goal GOAL-4 by backtracking

Value is (APPLICANT IS ENTITLED TO HEATING ADDITION OF 8.2)

Is this true?::

APPLICANT'S HOME HAS FIVE OR MORE ROOMS - EXCLUDING BATHROOM,  
LAVATORY AND HALL

:: no

The unnegated version of the condition PLANE-9 has not been proven.  
Therefore the condition (NIL APPLICANT IS ENTITLED TO HEATING ADDITION  
OF 8.2) is proven by default.

Current condition under consideration is (NIL APPLICANT IS ENTITLED TO  
HEATING ADDITION OF 5.05)

Trying to prove the goal GOAL-2 by backtracking

Value is (APPLICANT IS ENTITLED TO HEATING ADDITION OF 5.05)

Is this true?::

WARMTH IS REQUIRED DUE TO SERIOUS ILLNESS OF APPLICANT OR  
DEPENDANT - OR APPLICANT OR DEPENDANT IS HOUSEBOUND

:: no

Trying to prove the goal GOAL-2 by backtracking

Value is (APPLICANT IS ENTITLED TO HEATING ADDITION OF 5.05)

Is this true?::

APPLICANT'S HOME IS EXCEPTIONALLY DIFFICULT TO HEAT,  
PARTICULARLY DUE TO IT BEING VERY DAMP OR IN A VERY  
EXPOSED POSITION

:: yes

Trying to prove the goal GOAL-2 by backtracking

Value is (APPLICANT IS ENTITLED TO HEATING ADDITION OF 5.05)

The plane PLANE-12 is already known to be true.

Value is: (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 8.2)

Trying to prove the goal GOAL-2 by backtracking

Value is (APPLICANT IS ENTITLED TO HEATING ADDITION OF 5.05)

The plane PLANE-6 is already known to be true.

Value is: (NIL APPLICANT PAYS FOR HEATING IN RENT)

Trying to prove the goal GOAL-2 by backtracking

Value is (APPLICANT IS ENTITLED TO HEATING ADDITION OF 5.05)

The plane PLANE-7 is already known to be true.

Value is: (APPLICANT IS QUALIFIED AS CLAIMANT)

The current (true) condition (APPLICANT IS QUALIFIED AS CLAIMANT)  
is on the top level.

Therefore the goal (APPLICANT IS ENTITLED TO HEATING ADDITION OF 5.05)



has been proven. Returning to last downward search.  
The unnegated version of the condition PLANE-10 has been proven.  
Therefore the condition (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 5.05) is false.

Current condition under consideration is (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 8.2)

The plane PLANE-12 is already known to be true.

Value is: (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 8.2)

Current condition under consideration is (WARMTH IS REQUIRED DUE TO SERIOUS ILLNESS OF APPLICANT OR DEPENDANT - OR APPLICANT OR DEPENDANT IS HOUSEBOUND)

The plane PLANE-13 is already known to be false.

Value is: (WARMTH IS REQUIRED DUE TO SERIOUS ILLNESS OF APPLICANT OR DEPENDANT - OR APPLICANT OR DEPENDANT IS HOUSEBOUND)

Current condition under consideration is (APPLICANT'S HOME IS DIFFICULT TO HEAT ADEQUATELY, PARTICULARLY DUE TO ROOMS BEING DRAUGHTY, DAMP OR VERY LARGE)

Is this true?::

APPLICANT'S HOME IS DIFFICULT TO HEAT ADEQUATELY, PARTICULARLY  
DUE TO ROOMS BEING DRAUGHTY, DAMP OR VERY LARGE

:: no

Current condition under consideration is (APPLICANT'S HOME IS EXCEPTIONALLY DIFFICULT TO HEAT, PARTICULARLY DUE TO IT BEING VERY DAMP OR IN A VERY EXPOSED POSITION)

The plane PLANE-17 is already known to be true.

Value is: (APPLICANT'S HOME IS EXCEPTIONALLY DIFFICULT TO HEAT, PARTICULARLY DUE TO IT BEING VERY DAMP OR IN A VERY EXPOSED POSITION)

Current condition under consideration is (APPLICANT IS ENTITLED TO HEATING ADDITION OF 5.05)

The final condition in the downward search has been proven.

Therefore the goal GOAL-2 has been proven.

Value is (APPLICANT IS ENTITLED TO HEATING ADDITION OF 5.05)

Current condition under consideration is (APPLICANT'S HOME IS ON AN ESTATE CONSIDERED BY DHSS TO HAVE HIGH HEATING COSTS)

Is this true?::

APPLICANT'S HOME IS ON AN ESTATE CONSIDERED BY DHSS TO HAVE  
HIGH HEATING COSTS

:: no

Current condition under consideration is (APPLICANT OR DEPENDANTS IS RECEIVING MOBILITY OR ATTENDANCE ALLOWANCE OR EQUIVALENT PAID WITH A SERVICES OR DISABLEMENT PENSION)

Is this true?::

APPLICANT OR DEPENDANTS IS RECEIVING MOBILITY OR ATTENDANCE  
ALLOWANCE OR EQUIVALENT PAID WITH A SERVICES OR  
DISABLEMENT PENSION

:: no

Current condition under consideration is (APPLICANT'S HOUSEHOLD CONTAINS SOMEONE OVER 70 OR UNDER 5 YEARS OLD)

Is this true?::

APPLICANT'S HOUSEHOLD CONTAINS SOMEONE OVER 70 OR UNDER 5  
YEARS OLD

:: yes

Current condition under consideration is (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 5.05)

The plane PLANE-28 is already known to be false.

Value is: (NIL APPLICANT IS ENTITLED TO HEATING ADDITION OF 5.05)

Current condition under consideration is (APPLICANT'S HOME IS ON AN ESTATE CONSIDERED BY DHSS TO HAVE HIGH HEATING COSTS)

The plane PLANE-24 is already known to be false.

Value is: (APPLICANT'S HOME IS ON AN ESTATE CONSIDERED BY DHSS TO HAVE HIGH HEATING COSTS)

Current condition under consideration is (APPLICANT PAYS A SEPARATE FIXED CHARGE FOR HEATING, WHICH IS NOT SUBJECT TO REBATE OR SURCHARGE)

Is this true?::

APPLICANT PAYS A SEPARATE FIXED CHARGE FOR HEATING, WHICH IS  
NOT SUBJECT TO REBATE OR SURCHARGE

:: no

CC: information

Information on what: goals proven to be printed

The following goals have been proven:

: APPLICANT IS ENTITLED TO HEATING ADDITION OF 5.05

Information on what: ok

CC: ok

Setup, build or interpret:

APPENDIX C  
ANOTHER EXAMPLE - EUROPEAN COMMUNITY LAW

C.1 INTRODUCTION

It has been proposed that the techniques employed by the ELI project can be expanded into areas of the law apart from British Welfare Rights Legislation. It has also been proposed that the method of inputting rules into the ELI system makes it relatively easy to normalize and prepare rules from legislation.

In this Appendix we examine input dealing with the European Community sickness insurance scheme which was prepared by a legally qualified researcher (Helene Bauer-Bernet of the University Libre, Brussels and the Legal Service Department of the Commission of the European Communities); prior to producing these rules, Bauer-Bernet had seen one small demonstration of the system but had no access to it to aid in the preparation of her rule corpus. The legislation from which the rule corpus was prepared appears at the end of this section.

The inputting of the rules to the ELI system was carried out as follows, user input being underlined:

BUILD]

Current knowledge base or New or Old file? new file to hold new knowledge  
Name of file:: EEC.KNOW  
Delete any word and concept lists? yes  
Highest rule-number in KB is: 0  
BB: insert rules into knowledge base

Input request: from file  
Name file containing rules: EEC.RULE  
FILE CREATED 10-Jun-84 13:20:58  
RULESCOMS

17 rules in this file have not yet been input  
Rule numbering starts at: 1  
After how many rules have been input do you wish the knowledge base file  
and rule file to be updated: 20  
Extract words before inputting rules: Y/N: no  
Words will be extracted as rules are input  
I1I2G3I4G5I6T7G8G9I10I11I12I13I14T15I16T17  
Knowledge base being updated to: EEC.KNOW  
Writing to EEC.RULE  
Done.

BB:print rules from knowledge base

Print request: T  
Print in which order (R)ule, (K)nowledge base or (OK): Knowledge base order  
Writing in KB order: (FILECREATED "10-Jun-84 13:48:33" T)

- 1 (IF ((A PERSON IS COVERED))  
THEN  
(THE INSURANCE SCHEME WILL GUARANTEE TO THE PERSON REIMBURSEMENT OF  
EXPENSES INCURRED AS A RESULT OF ILLNESS, ACCIDENT OR CONFINEMENT  
AND THE PAYMENT OF AN ALLOWANCE TOWARDS FUNERAL EXPENSES))
- 2 (IF ((THE PERSON IS A MEMBER))  
THEN  
(THE PERSON IS COVERED))
- 3 (IF ((THE PERSON IS COVERED BY THE INSURANCE OF A MEMBER))  
THEN  
(THE PERSON IS COVERED))
- 4 (IF ((THE PERSON IS AN OFFICIAL OF THE COMMUNITIES)  
(NIL THE PERSON IS ON LEAVE ON PERSONAL GROUNDS - UNDER ARTICLE 40 OF  
THE STAFF REGULATIONS, SUBJECT TO ARTICLE 4{1} OF THESE RULES))  
THEN  
(THE PERSON IS A MEMBER))
- 5 (IF ((THE PERSON IS PRESIDENT, VICE-PRESIDENT OR MEMBER OF THE COMMISSION  
OF THE EUROPEAN COMMUNITIES, OR PRESIDENT, JUDGE,  
ADVOCATE-GENERAL OR REGISTRAR OF THE COURT OF JUSTICE OF THE  
EUROPEAN COMMUNITIES))  
THEN  
(THE PERSON IS A MEMBER))
- 6 (IF ((THE PERSON IS THE SPOUSE)  
(NIL THE PERSON IS A MEMBER)  
(NIL S/HE IS GAINFULLY EMPLOYED))  
THEN  
(THE PERSON IS COVERED BY THE MEMBER'S INSURANCE))
- 7 (IF ((THE PERSON IS THE SPOUSE)  
(NIL HIS/HER ANNUAL INCOME FROM SUCH EMPLOYMENT EXCEEDS BF 500,000  
BEFORE TAX)  
(S/HE IS COVERED BY A PUBLIC SCHEME OF SICKNESS INSURANCE)  
(S/HE IS GAINFULLY EMPLOYED)  
(NIL S/HE IS A MEMBER))  
THEN  
(THE PERSON IS COVERED BY THE MEMBER'S INSURANCE))
- 8 (IF ((THE PERSON IS A DEPENDANT CHILD WITHIN THE MEANING OF STAFF  
REGULATIONS, ANNEX VII, ART.2{2}))  
THEN

- (THE PERSON IS COVERED BY THE MEMBER'S INSURANCE))
- 9 (IF ((NIL THE PERSON CAN OBTAIN COVER UNDER ANY OTHER PUBLIC SCHEME OF SICKNESS INSURANCE)  
(THE PERSON IS TREATED AS A DEPENDANT CHILD OF THE MEMBER PURSUANT TO THE STAFF REGULATIONS, ANNE EX VII, ART.2{4}))  
THEN  
(THE PERSON IS COVERED BY THE MEMBER'S INSURANCE))
- 10 (IF ((A PERSON IS MEMBER OF THE TEMPORARY STAFF)  
(THE PERSON IS SUBJECT TO A RESTRICTION FOR THE REIMBURSEMENT OF CERTAIN EXPENSES {CF. 7.1})  
(THE PERSON IS MEMBER OF THE TEMPORARY STAFF SINCE TWO YEARS OR MORE)  
(THE MEDICAL OFFICER OF THE INSTITUTION IS OF THE OPINION THAT THE SICKNESS OR INVALIDITY HAS NOT REAPPEARED OR GIVEN RISE TO UNUSUAL SEQUELAE IN THE COURSE OF THE SAID PERIOD))  
THEN  
(THE COMPETENT AUTHORITY {= THE AUTHORITY EMPOWERED TO CONCLUDE CONTRACTS OF EMPLOYMENT ON BEHALF OF THE INSTITUTION TO WHICH THE PERSON CONCERNED BELONGS} MAY RAISE THE RESTRICTION))
- 11 (IF ((AN ACCIDENT OF ILLNESS IS CAUSED BY A THIRD PARTY TO A PERSON COVERED))  
THEN  
(THE RIGHT OF ACTION OF THE PERSON CONCERNED OR OF THOSE ENTITLED UNDER HIM AGAINST THE THIRD PARTY SHALL REST IN THE COMMUNITIES))
- 12 (IF ((NIL A MEMBER HAS APPLIED FOR REIMBURSEMENT OF EXPENSES INCURRED BY HIM OR BY A PERSON COVERED BY THE MEMBER'S INSURANCE DURING THE CALENDAR YEAR FOLLOWING THAT IN WHICH TREATMENT WAS ADMINISTERED)  
(NIL FORCE MAJEURE IS DULY ESTABLISHED))  
THEN  
(NIL THE MEMBER IS ENTITLED TO REIMBURSEMENT))
- 13 (IF ((NIL THE MEMBER APPLIES FOR SPECIAL REIMBURSEMENT REFERRED TO IN ART. 8{2} WITHIN 24 MONTHS OF THE DATE ON WHICH THE EXPENSES LAST INCURRED IN RESPECT OF TREATMENT WITHIN THE TWELVE-MONTH PERIOD IN QUESTION WERE REIMBURSED))  
THEN  
(NIL THE MEMBER IS ENTITLED TO SPECIAL REIMBURSEMENT))
- 14 (IF ((THE PERSON RECEIVES UNDUE PAYMENT)  
(THE PERSON WAS AWARE THAT THERE WAS NO DUE REASON FOR THE PAYMENT))  
THEN  
(THE SUM OVERPAID SHALL BE RECOVERED))
- 15 (IF ((THE PERSON RECEIVES UNDUE PAYMENT)  
(THE FACT OF THE OVERPAYMENT WAS PATENTLY SUCH))  
THEN  
(THE SUM OVERPAID SHALL BE RECOVERED))
- 16 (IF ((A MEMBER FRAUDULENTLY OBTAINS OR ATTEMPTS TO OBTAIN BENEFITS FOR HIMSELF OR FOR A PERSON COVERED BY THE MEMBER'S INSURANCE))  
THEN  
(NIL THE PERSON IS ENTITLED TO SUCH BENEFITS))
- 17 (IF ((A MEMBER FRAUDULENTLY OBTAINS OR ATTEMPTS TO OBTAIN BENEFITS FOR HIMSELF OR FOR A PERSON COVERED BY THE MEMBER'S INSURANCE))  
THEN  
(THE PERSON SHALL BE LIABLE TO DISCIPLINARY ACTION))

Only some parts of the legislation have been normalized, since this was undertaken as a test of the system; thus, for example, in Article 2, (which deals with the description of those who are members of the Sickness Insurance Scheme) only the first and final clauses have been dealt with. Obviously, the remaining clauses are easily handled if these can be. Rules 4 and 5 are formalizations of these membership clauses.

An example interaction with this rule base is as follows:

(INTERROGATE)

Current or new knowledge base: new knowledge base to be loaded

Name of file: EEC.KNOW

FILE CREATED 10-Jun-84 15:33:01

KNOWBASECOMS

CC: initialize

CC: start interpretation

Is this true?::

THE PERSON IS ON LEAVE ON PERSONAL GROUNDS - UNDER ARTICLE 40 OF THE  
STAFF REGULATIONS, SUBJECT TO ARTICLE 4{1} OF THESE RULES

:: no

Is this true?::

THE PERSON IS A AN OFFICIAL OF THE COMMUNITIES

:: yes

Is this true?::

THE PERSON IS THE SPOUSE

:: no

Is this true?::

THE PERSON IS A DEPENDANT CHILD WITHIN THE MEANING OF STAFF  
REGULATIONS, ANNEX VII, ART.2{2}

:: no

Is this true?::

THE PERSON CAN OBTAIN COVER UNDER ANY OTHER PUBLIC SCHEME OF SICKNESS  
INSURANCE

:: yes

Is this true?::

A PERSON IS MEMBER OF THE TEMPORARY STAFF

:: yes

Is this true?::

THE PERSON IS SUBJECT TO A RESTRICTION FOR THE REIMBURSEMENT OF  
CERTAIN EXPENSES {CF. 7.1}

:: no

Is this true?::

AN ACCIDENT OF ILLNESS IS CAUSED BY A THIRD PARTY TO A PERSON COVERED

:: yes

Is this true?::

A MEMBER HAS APPLIED FOR REIMBURSEMENT OF EXPENSES INCURRED BY HIM OR  
BY A PERSON COVERED BY THE MEMBER'S INSURANCE DURING THE CALENDAR  
YEAR FOLLOWING THAT IN WHICH TREATMENT WAS ADMINISTERED

:: yes

Is this true?::

THE MEMBER APPLIES FOR SPECIAL REIMBURSEMENT REFERRED TO IN ART. 8{2}  
WITHIN 24 MONTHS OF THE DATE ON WHICH THE EXPENSES LAST INCURRED

IN RESPECT OF TREATMENT WITHIN THE TWELVE-MONTH PERIOD IN  
QUESTION WERE REIMBURSED

:: no

Is this true?::

THE PERSON RECEIVES UNDUE PAYMENT

:: yes

Is this true?::

THE PERSON WAS AWARE THAT THERE WAS NO DUE REASON FOR THE PAYMENT

:: no

Is this true?::

THE FACT OF THE OVERPAYMENT WAS PATENTLY SUCH

:: no

Is this true?::

A MEMBER FRAUDULENTLY OBTAINS OR ATTEMPTS TO OBTAIN BENEFITS FOR  
HIMSELF OR FOR A PERSON COVERED BY THE MEMBER'S INSURANCE

:: no

CC: Information

Information on what: known facts to be printed

The following items are known:

- : NIL THE PERSON IS ON LEAVE ON PERSONAL GROUNDS - UNDER ARTICLE 40 OF  
THE STAFF REGULATIONS, SUBJECT TO ARTICLE 4 (1) OF THESE RULES
- : THE PERSON IS A AN OFFICIAL OF THE COMMUNITIES
- : THE PERSON IS A MEMBER
- : THE PERSON IS COVERED
- : THE INSURANCE SCHEME WILL GUARANTEE TO THE PERSON REIMBURSEMENT OF  
EXPENSES INCURRED AS A RESULT OF ILLNESS, ACCIDENT OR CONFINEMENT  
AND THE PAYMENT OF AN ALLOWANCE TOWARDS FUNERAL EXPENSES
- : A PERSON IS MEMBER OF THE TEMPORARY STAFF
- : AN ACCIDENT OF ILLNESS IS CAUSED BY A THIRD PARTY TO A PERSON COVERED
- : THE RIGHT OF ACTION OF THE PERSON CONCERNED OR OF THOSE ENTITLED  
UNDER HIM AGAINST THE THIRD PARTY SHALL REST IN THE COMMUNITIES
- : NIL THE MEMBER APPLIES FOR SPECIAL REIMBURSEMENT REFERRED TO IN ART.  
8{2} WITHIN 24 MONTHS OF THE DATE ON WHICH THE EXPENSES LAST  
INCURRED IN RESPECT OF TREATMENT WITHIN THE TWELVE-MONTH PERIOD  
IN QUESTION WERE REIMBURSED
- : NIL THE MEMBER IS ENTITLED TO SPECIAL REIMBURSEMENT
- : THE PERSON RECEIVES UNDUE PAYMENT

Information on what: not true facts are to be printed

The following items are known not to be true:

- : THE PERSON IS THE SPOUSE
- : THE PERSON IS A DEPENDANT CHILD WITHIN THE MEANING OF STAFF  
REGULATIONS, ANNEX VII, ART.2{2}
- : NIL THE PERSON CAN OBTAIN COVER UNDER ANY OTHER PUBLIC SCHEME OF  
SICKNESS INSURANCE
- : THE PERSON IS SUBJECT TO A RESTRICTION FOR THE REIMBURSEMENT OF  
CERTAIN EXPENSES {CF. 7.1}
- : NIL A MEMBER HAS APPLIED FOR REIMBURSEMENT OF EXPENSES INCURRED BY HIM  
OR BY A PERSON COVERED BY THE MEMBER'S INSURANCE DURING THE  
CALENDAR YEAR FOLLOWING THAT IN WHICH TREATMENT WAS ADMINISTERED
- : THE PERSON WAS AWARE THAT THERE WAS NO DUE REASON FOR THE PAYMENT
- : THE FACT OF THE OVERPAYMENT WAS PATENTLY SUCH
- : A MEMBER FRAUDULENTLY OBTAINS OR ATTEMPTS TO OBTAIN BENEFITS FOR  
HIMSELF OR FOR A PERSON COVERED BY THE MEMBER'S INSURANCE

## C.2 DISCUSSION

### C.2.1 Conceptual Matching

The first point to be noted is that in the small test corpus, there was no use made of the important word or noise word lists - those lists which respectively highlight or remove words from the input conditions to aid the conceptual matching module of ELI. The absence of these words made no difference to the inputting of the rules into the knowledge base in the BUILD process (as seen above); what does occur though is that in the first interpretations, the system finds similar words in the conditions/goals (for example, "PERSON") and asks the user whether these match. This can become, after a little while, quite tedious:

```
(THE PERSON IS COVERED)
(A PERSON IS COVERED)
Do these match :: yes
(NIL THE MEMBER IS ENTITLED TO REIMBURSEMENT)
(NIL THE PERSON IS ON LEAVE ON PERSONAL GROUNDS - UNDER ARTICLE 40 OF THE
STAFF REGULATIONS, SUBJECT TO ARTICLE 4{1} OF THESE RULES)
Do these match :: no
Is this true?::
    THE PERSON IS ON LEAVE ON PERSONAL GROUNDS - UNDER ARTICLE 40 OF THE
        STAFF REGULATIONS, SUBJECT TO ARTICLE 4 (1) OF THESE RULES
:: no
Is this true?::
    THE PERSON IS A AN OFFICIAL OF THE COMMUNITIES
:: yes
(THE PERSON IS A AN OFFICIAL OF THE COMMUNITIES)
(A PERSON IS COVERED)
Do these match :: no
(THE PERSON IS A MEMBER)
(A PERSON IS COVERED)
Do these match :: no
(THE PERSON IS A AN OFFICIAL OF THE COMMUNITIES)
(THE PERSON IS THE SPOUSE)
Do these match :: no
(THE PERSON IS A MEMBER)
(THE PERSON IS THE SPOUSE)
Do these match :: no
(THE PERSON IS COVERED)
(THE PERSON IS THE SPOUSE)
Do these match :: no
```

Another point related to the operation of the system is that in some of the rules there is a slight difference of format. For example, rule 1 has the condition, "A PERSON IS COVERED" and rules 2 and 3 have the goal



"THE PERSON IS COVERED". These slight aberrations (and many more serious ones) must be expected as normal. A system which operated purely upon the basis of exact textual matching would not recognize the condition of rule 1 as matching the goals of rules 2 and 3, but the ELI system does.

Currently the ELI system carries out the processing for this conceptual matching during the BUILD or INTERPRET stage, rather than in the SETUP stage. The reason for doing it this way was that the originally envisaged manner of inputting the rules was to develop them on paper and then input them to the system. It is now becoming clear that perhaps more processing should be put into the SETUP operation - so that, in fact, the SETUP module interactively aids the expert in the normalization process. The type of activities which it might carry out are those administrative ones, including indexing of rules where specified words are contained, the segmentation of the entire corpus into smaller more conceptualizable blocks of rules, and the keeping of aides-memoire - rather than the current practice of making little notes on odd pieces of paper.

Such a module would be very much more complex than the current one, but we would suggest that in any commercially viable system such a complex but user directed module is essential.

### C.2.2 The Use Of Negation

The test corpus provided by Bauer-Bernet made no use of that form of negation which the system cannot understand (that is, where "not" is included within the text of the condition/goal); only that negation of conditions/goals which is introduced by the prefix "NIL" and which is 'recognized' by the system is used. This tends to imply that such inclusion of 'artificial' english within the production rules is not too difficult. When preparing the original rules for the ELI system (those

dealing with Supp. Ben.) it was sometimes inadvertently forgotten that it was better to use the prefixed negation and include it within the text. When glancing at the rules though, it was easy to realize what had happened and amend the conditions/goals to the preferred format. We would suggest that this might well be how others would do it as well.

One aspect of the use of negation in the system which becomes relatively easy to use - when some practice and experience with the system has been achieved - is the pruning of rules from the knowledge base. This, of course, requires rules designed specifically to do this rather than rules which are a direct normalization of the legislation. The pruning of rules from the knowledge base leads to quicker interaction with the system (because "unnecessary" questions are not being asked) and less frustration with the system (once again, because unnecessary questions are not being asked).

As an example of this, take the case where the person in question is not a member nor covered by a member's insurance. The interpretation in this case might look something like this:

```
Current or new knowledge base: new knowledge base to be loaded
Name of file: EEC.KNOW
FILE CREATED 12-Jun-84 21:43:12
KNOWBASECOMS
CC: initialize
CC: start interpretation
Is this true?::
    THE PERSON IS ON LEAVE ON PERSONAL GROUNDS - UNDER ARTICLE 40 OF THE
    STAFF REGULATIONS, SUBJECT TO ARTICLE 4{1} OF THESE RULES
:: no
Is this true?::
    THE PERSON IS A AN OFFICIAL OF THE COMMUNITIES
:: no
Is this true?::
    THE PERSON IS PRESIDENT, VICE-PRESIDENT OR MEMBER OF THE COMMISSION OF
    THE EUROPEAN COMMUNITIES, OR PRESIDENT, JUDGE, ADVOCATE-GENERAL
    OR REGISTRAR OF THE COURT OF JUSTICE OF THE EUROPEAN COMMUNITIES
:: no
Is this true?::
    THE PERSON IS COVERED BY THE INSURANCE OF A MEMBER
:: no
Is this true?::
    THE PERSON IS THE SPOUSE
:: no
Is this true?::
    THE PERSON IS A DEPENDANT CHILD WITHIN THE MEANING OF STAFF
```

```

:: no
Is this true?::
    THE PERSON CAN OBTAIN COVER UNDER ANY OTHER PUBLIC SCHEME OF SICKNESS
    INSURANCE
:: yes
Is this true?::
    A PERSON IS MEMBER OF THE TEMPORARY STAFF
:: no
Is this true?::
    AN ACCIDENT OF ILLNESS IS CAUSED BY A THIRD PARTY TO A PERSON COVERED
:: no
Is this true?::
    A MEMBER HAS APPLIED FOR REIMBURSEMENT OF EXPENSES INCURRED BY HIM OR
    BY A PERSON COVERED BY THE MEMBER'S INSURANCE DURING THE CALENDAR
    YEAR FOLLOWING THAT IN WHICH TREATMENT WAS ADMINISTERED
:: no
Is this true?::
    FORCE MAJEURE IS DULY ESTABLISHED
:: no
Is this true?::
    THE MEMBER APPLIES FOR SPECIAL REIMBURSEMENT REFERRED TO IN ART. 8{2}
    WITHIN 24 MONTHS OF THE DATE ON WHICH THE EXPENSES LAST INCURRED
    IN RESPECT OF TREATMENT WITHIN THE TWELVE-MONTH PERIOD IN
    QUESTION WERE REIMBURSED
:: no
Is this true?::
    THE PERSON RECEIVES UNDUE PAYMENT
:: no
Is this true?::
    A MEMBER FRAUDULENTLY OBTAINS OR ATTEMPTS TO OBTAIN BENEFITS FOR
    HIMSELF OR FOR A PERSON COVERED BY THE MEMBER'S INSURANCE
:: no
CC: Information
    Information on what: goals proven to be printed
    The following goals have been proven:
    : NIL THE MEMBER IS ENTITLED TO REIMBURSEMENT
    : NIL THE MEMBER IS ENTITLED TO SPECIAL REIMBURSEMENT

```

There are still several questions being asked which are not relevant after the system has received advice. It would be desirable to allow the system to remove the rules which drive these questions when it was known by the system that it was pointless to ask them.

In the ELI system this meta-level control of rule invocation is done either by using a proven goal to remove its negated or non-negated corresponding goal from the knowledge base, or by having a top-level condition which, when disproven, removes all rules which have it as their top condition from the knowledge base. As an example of the first, proving the goal (NIL this is today) will remove from the knowledge base all rules whose goal is (this is today). As example of

the second is where the first condition met is (today is friday); if this is untrue, then all rules with it as their top condition are removed from the knowledge base.

This latter kind of meta-control is used frequently in the ELI Supp. Ben. rules - by creating the notion of "applicant satisfies rules of entitlement to supplementary benefit" (which is not a truly "legal" concept) and using this as the top level condition of many rules then whenever the condition is proven false all these rules are removed from the knowledge base.

In effect, what is being said is that not all the rules within the knowledge base must necessarily be direct representations of normalized legislation; in order to aid speed and fluency of interpretation other rules can be easily added.

### C.2.3 Legislative Self-reference

Bauer-Bernet's rule corpus indicates one of the aspects of the production system formalism which is particularly disadvantageous for legal advisory systems - that of the need for the rules to encapsulate more than the information required for interpretation. Analogically, we can see that this need arises from what Bar-Hillel [Bar-Hillel, 1964] has noted as the two types of information retrieval - that of the retrieval of facts and the retrieval of reference. The first type of fact retrieval we can see as that which the interpreter requires for processing - it needs to know whether a given condition or goal is false or true. The second type of retrieval is that where the rules refer, not to other rules, but to specific parts of the legislation. The second condition of rule number 4, above, is an example of this:

NIL the person is on leave on personal grounds - under article 40 of the staff regulations, subject to article 4(1) of these rules)

This condition could be broken down into more basic rules, but doing so removes much information which is understood by someone with experience in the legislative domain. By producing more production rules from it, there is a danger of "seeing the trees, but not the wood".

This is a disadvantage because we can expect the users of such a system to have some legal competence, whether they are legally qualified or not. There may be some occasions when a complete step by step interpretation through the rules is necessary, but on many occasions the user will have the ability to short-circuit the interpretation by, in effect, saying - "yes, the person is not on leave according to that specified article".

It seems that one future area of research is in the area of using the system's annotation facility to encapsulate information about where the rule is leading, and then the user has the option of continuing with the interpretation or terminating that line of enquiry. Such a facility would be useful as a means of making the system much more user friendly than it currently is.

#### C.2.4 Difficulties Found In Normalization

The two main difficulties in normalizing the legislation found by Bauer-Bernet were those in providing a formulation for Article 11 on procedure and Article 16 on appeals.

It may be that the problems found were due to the fact that these two articles concern procedures rather than "legal rules"; that is, they do not define legal concepts or memberships of legal groups - only the procedures which must be followed in certain circumstances. This supports our view that the production system formalism is not sufficient for useful legal systems; even though it is possible that these articles could be normalized. For example, one possible manner to handle Article 11 is:

IF (application for prior authorization is required)

(prior authorization has been given)

(correct procedures for prior authorization have been followed)

THEN

(application for prior authorization is allowed)

IF (the application for prior authorization, together with a prescription and/or an estimate made out by the attending practitioner, has been submitted by the member to the office responsible for settling claims, which has referred the matter to its medical officer if required. In the latter case, the medical officer has transmitted his Opinion to the office responsible for settling claims within two weeks)  
(the office responsible for settling claims has taken a decision on the application if it has been appointed to do so or has transmitted its decision and, where applicable, that of the medical officer to the appointing authority for a decision. The member has then been informed of the decision forthwith.)

THEN

(correct procedures for prior authorization have been followed)

This latter rule could have been broken down into smaller rules, and can, of course, have annotations added.

One interesting point which this example brings to light is that the manner in which the rules are prepared must be designed with the eventual user in mind. Thus if the system is designed to allow interaction with a member, it will present different advice from a system which is designed to be used by someone involved in the administration of the scheme. Niblett [Niblett, 1981] has pointed out that these expert systems will have different views of law and different potential users; we would agree with him.

### C.2.5 Presentations Of Conclusions

As can be seen by the example interpretation above, the system simply tries to effect all goals in the knowledge base and only terminates when they there are no more rules left to test. At that point, the user can ask for information about the goals which have been proven and about conditions and goals which are true or false.

Such an attempt to prove all goals is important in the welfare rights system - because, after all, the more goals (that is, monetary or other benefits) which can be achieved the better.

Other legislative areas are not appropriate for this type of processing - a much better strategy is to assert the specific goal which we wish to discover is true or not:

Is it true that the person is a member of the sickness scheme?

or alternatively to specify a stopping position:

Stop when it is proven that the person is not a member of the sickness scheme.

But such strategies are relatively easy to implement, using a user interface between the user and the program. Each implementation of a legal expert system, we would suggest, would require an engineered interface designed specifically for each type of application.

## APPENDIX D

### A LISTING OF PART OF THE ELI KNOWLEDGE BASE

In this appendix we list 50 rules from the knowledge base. These rules were not chosen to demonstrate any particular aspects - they are simply the first fifty in the knowledge base and are thus representative of the system rules. The remaining rules are omitted for reasons of space, but can be listed from the ELI program.

- RULE-1 (IF ((APPLICANT IS AVAILABLE FOR WORK)  
(APPLICANT HAS A WORK PERMIT))  
THEN  
(NIL APPLICANT SATISFIES RULES OF ENTITLEMENT TO  
SUPPLEMENTARY BENEFIT))
- RULE-2 (IF ((APPLICANT IS AVAILABLE FOR WORK)  
(APPLICANT IS A BRITISH CITIZEN))  
THEN  
(APPLICANT SATISFIES RULES OF ENTITLEMENT TO  
SUPPLEMENTARY BENEFIT))
- RULE-3 (IF ((APPLICANT IS AVAILABLE FOR WORK)  
(APPLICANT IS NATIONAL OF COMMON MARKET COUNTRY))  
THEN  
(APPLICANT SATISFIES RULES OF ENTITLEMENT TO  
SUPPLEMENTARY BENEFIT))
- RULE-4 (IF ((APPLICANT IS AVAILABLE FOR WORK)  
(APPLICANT IS A NATIONAL OF ICELAND, MALTA, NORWAY,  
PORTUGAL, SWEDEN OR TURKEY))  
THEN  
(APPLICANT SATISFIES RULES OF ENTITLEMENT TO  
SUPPLEMENTARY BENEFIT))
- RULE-5 (IF ((APPLICANT IS AVAILABLE FOR WORK)  
(APPLICANT HAS BEEN GRANTED FOREIGN HUSBAND STATUS,  
REFUGEE STATUS OR POLITICAL ASYLUM))  
THEN  
(APPLICANT SATISFIES RULES OF ENTITLEMENT TO  
SUPPLEMENTARY BENEFIT))
- RULE-6 (IF ((APPLICANT IS AVAILABLE FOR WORK)  
(APPLICANT HAS RIGHT OF ABODE IN UK))  
THEN  
(APPLICANT SATISFIES RULES OF ENTITLEMENT TO  
SUPPLEMENTARY BENEFIT))
- RULE-7 (IF ((APPLICANT IS AVAILABLE FOR WORK)  
(APPLICANT HAS CERTIFICATE OF PATRIALITY OR A



CERTIFICATE OF ENTITLEMENT TO ABODE FROM THE  
HOME OFFICE))

- THEN  
(APPLICANT SATISFIES RULES OF ENTITLEMENT TO  
SUPPLEMENTARY BENEFIT))
- RULE-8 (IF ((APPLICANT IS AVAILABLE FOR WORK)  
(APPLICANT HAS BEEN GRANTED INDEFINITE LEAVE TO ENTER OR  
RESIDE IN THE UK))
- THEN  
(APPLICANT SATISFIES RULES OF ENTITLEMENT TO  
SUPPLEMENTARY BENEFIT))
- RULE-9 (IF ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT)  
(APPLICANT IS A BRITISH OVERSEAS CITIZEN WITH RIGHT OF  
READMISSION TO THE UK))
- THEN  
(APPLICANT SATISFIES RULES OF ENTITLEMENT TO  
SUPPLEMENTARY BENEFIT))
- RULE-10 (IF ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT)  
(APPLICANT OVER NORMAL AGE OF RETIREMENT))
- THEN  
(SUPPLEMENTARY BENEFIT IS KNOWN AS SUPPLEMENTARY PENSION))
- RULE-11 (IF ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT)  
(APPLICANT UNDER NORMAL AGE OF RETIREMENT)  
(APPLICANT IS OVER 16 YEARS OLD))
- THEN  
(SUPPLEMENTARY BENEFIT IS KNOWN AS SUPPLEMENTARY  
ALLOWANCE))
- RULE-12 (IF ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT))
- THEN  
(APPLICANT ENTITLED TO HOUSING BENEFIT))
- RULE-13 (IF ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT)  
(APPLICANT'S IS INCAPABLE FOR WORK THROUGH ILLNESS OR  
DISABILITY)  
(APPLICANT'S EVIDENCE OF INCAPACITY IS ACCEPTED BY DHSS))
- THEN  
(APPLICANT NEED NOT SIGN AS AVAILABLE FOR WORK))
- RULE-14 (IF ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT)  
(APPLICANT CANNOT WORK FULL-TIME BECAUSE OF DISEASE OR  
DISABLEMENT))
- THEN  
(APPLICANT NEED NOT SIGN AS AVAILABLE FOR WORK))
- RULE-15 (IF ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT)  
(APPLICANT IS LOOKING AFTER A SEVERELY DISABLED PERSON  
WHO CANNOT BE LOOKED AFTER ANY OTHER WAY))
- THEN  
(APPLICANT NEED NOT SIGN AS AVAILABLE FOR WORK))
- RULE-16 (IF ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT)  
(APPLICANT IS BLIND)  
(APPLICANT HAS BEEN USED TO WORKING OUTSIDE THE HOME FOR  
THE LAST 12 MONTHS))
- THEN  
(APPLICANT NEED NOT SIGN AS AVAILABLE FOR WORK))
- RULE-17 (IF ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT)  
(APPLICANT IS FEMALE)  
(APPLICANT IS PREGNANT)  
(APPLICANT HAS A MEDICAL CERTIFICATE OF INABILITY TO  
WORK))
- THEN  
(APPLICANT NEED NOT SIGN AS AVAILABLE FOR WORK))
- RULE-18 (IF ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT)  
(APPLICANT IS FEMALE)  
(APPLICANT IS DUE TO GIVE BIRTH IN LESS THAN 12 WEEKS))
- THEN

- RULE-19 (IF ((APPLICANT NEED NOT SIGN AS AVAILABLE FOR WORK))  
 ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT)  
 (APPLICANT HAS A PHYSICAL OR MENTAL DISABILITY CAUSING  
 NO FURTHER PROSPECTS OF FINDING WORK)  
 (APPLICANT HAS WORKED LESS THAN 4 HOURS PER WEEK FOR THE  
 LAST 12 WEEKS)  
 (APPLICANT HAS SINGED ON FOR AT LEAST 39 WEEKS)  
 (APPLICANT HAS MADE REASONABLE EFFORTS TO FIND WORK)  
 (APPLICANT HAS NOT REFUSED SUITABLE WORK))  
 THEN  
 (APPLICANT NEED NOT SIGN AS AVAILABLE FOR WORK))
- RULE-20 (IF ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT)  
 (APPLICANT IS WITHIN 10 YEARS OF RETIREMENT AGE)  
 (APPLICANT HAS NO PROSPECT OF FUTURE EMPLOYMENT BECAUSE  
 OF LACK OF TRAINING OR EXPERIENCE)  
 (APPLICANT HAS BEEN UNEMPLOYED FOR 10 YEARS))  
 THEN  
 (APPLICANT NEED NOT SIGN AS AVAILABLE FOR WORK))
- RULE-21 (IF ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT)  
 (APPLICANT IS ATTENDING AN OPEN UNIVERSITY COURSE))  
 THEN  
 (APPLICANT NEED NOT SIGN AS AVAILABLE FOR WORK))
- RULE-23 (IF ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT)  
 (APPLICANT RECEIVES A JOB RELEASE SCHEME ALLOWANCE))  
 THEN  
 (APPLICANT NEED NOT SIGN AS AVAILABLE FOR WORK))
- RULE-24 (IF ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT)  
 (APPLICANT IS A REFUGEE)  
 (APPLICANT HAS LIVED IN BRITAIN FOR LESS THAN A YEAR)  
 (APPLICANT IS LEARNING ENGLISH IN A PART-TIME COURSE))  
 THEN  
 (APPLICANT NEED NOT SIGN AS AVAILABLE FOR WORK))
- RULE-25 (IF ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT)  
 (APPLICANT MUST ATTEND COURSE FOR MORE THAN 2 DAYS AS  
 JP, JUROR, WITNESS OR PARTY TO PROCEEDINGS))  
 THEN  
 (APPLICANT NEED NOT SIGN AS AVAILABLE FOR WORK))
- RULE-26 (IF ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT)  
 (APPLICANT HAS BEEN REMANDED IN CUSTODY OR COMMITTED IN  
 CUSTODY FOR TRIAL OR TO BE SENTENCED))  
 THEN  
 (APPLICANT NEED NOT SIGN AS AVAILABLE FOR WORK))
- RULE-27 (IF ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT)  
 (APPLICANT IS A DISABLED STUDENT)  
 (APPLICANT'S DISABILITY DEBARS HIM/HER FROM GETTING WORK  
 AS EASILY AS OTHERS))  
 THEN  
 (APPLICANT NEED NOT SIGN AS AVAILABLE FOR WORK))
- RULE-28 (IF ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT)  
 (APPLICANT IS A STUDENT)  
 (APPLICANTS UNMARRIED PARTNER IS A NON-STUDENT)  
 (APPLICANT'S PARTNER HAS BEEN ILL FOR 8 WEEKS))  
 THEN  
 (APPLICANT NEED NOT SIGN AS AVAILABLE FOR WORK))
- RULE-29 (IF ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT)  
 (APPLICANT IS A STUDENT)  
 (APPLICANT'S PARTNER HAS A CHILD IS WHICH IS NOT  
 APPLICANTS)  
 (APPLICANTS'S UNMARRIED PARTNER IS A NON-STUDENT))  
 THEN  
 (APPLICANT NEED NOT SIGN AS AVAILABLE FOR WORK))
- RULE-30 (IF ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT)  
 (APPLICANT IS 19 YEARS OLD)

(APPLICANT IS ON A NON-ADVANCED COURSE))  
 THEN  
 (APPLICANT NEED NOT SIGN AS AVAILABLE FOR WORK))  
 RULE-31 (IF ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT)  
 (APPLICANT IS A FULL-TIME SCHOOL STUDENT))  
 THEN  
 (APPLICANT NEED NOT SIGN AS AVAILABLE FOR WORK))  
 RULE-32 (IF ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT)  
 (APPLICANT IS ON STRIKE))  
 THEN  
 (APPLICANT NEED NOT SIGN AS AVAILABLE FOR WORK))  
 RULE-33 (IF ((APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT)  
 (APPLICANT HAS RETURNED TO WORK AFTER STRIKE))  
 THEN  
 (APPLICANT NEED NOT SIGN AS AVAILABLE FOR WORK))  
 RULE-34 (IF ((APPLICANT SATISFIES RULES OF ENTITLEMENT TO  
 SUPPLEMENTARY BENEFIT)  
 (APPLICANT IS OVER 16 YEARS OLD)  
 (APPLICANT IS USUALLY RESIDENT IN GREAT BRITAIN)  
 (NIL APPLICANT IS IN FULL-TIME WORK))  
 THEN  
 (APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT))  
 RULE-35 (IF ((APPLICANT IS RECEIVING A CATEGORY A RETIREMENT PENSION)  
 (APPLICANT OVER NORMAL AGE OF RETIREMENT))  
 THEN  
 (APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT))  
 RULE-36 (IF ((APPLICANT RETIRED FROM FULL-TIME WORK ON OR AFTER  
 REACHING PENSIONABLE AGE)  
 (APPLICANT OVER NORMAL AGE OF RETIREMENT))  
 THEN  
 (APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT))  
 RULE-37 (IF ((APPLICANT RETIRED IN THE FIVE YEARS BEFORE REACHING  
 PENSIONABLE AGE)  
 (APPLICANT OVER NORMAL AGE OF RETIREMENT))  
 THEN  
 (APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT))  
 RULE-38 (IF ((NIL APPLICANT IS DISABLED)  
 (APPLICANTS WORKS 30 OR MORE HOURS PER WEEK)  
 (NIL APPLICANT SATISFIES FULL-TIME WORK EXCEPTIONS))  
 THEN  
 (APPLICANT IS IN FULL-TIME WORK))  
 RULE-39 (IF ((NIL APPLICANT IS DISABLED)  
 (APPLICANT WORKS LESS THAN 30 HOURS PER WEEK))  
 THEN  
 (APPLICANT IS IN PART-TIME WORK))  
 RULE-40 (IF ((APPLICANT IS DISABLED)  
 (APPLICANT WORKS 35 OR MORE HOURS PER WEEK)  
 (NIL APPLICANT SATISFIES FULL-TIME WORK EXCEPTIONS))  
 THEN  
 (APPLICANT IS IN FULL-TIME WORK))  
 RULE-41 (IF ((APPLICANT IS DISABLED)  
 (APPLICANT WORKS LESS THAN 35 HOURS PRE WEEK))  
 THEN  
 (APPLICANT IS IN PART-TIME WORK))  
 RULE-42 (IF ((APPLICANT IS OFF WORK DUE TO A RECOGNIZED OR CUSTOMARY  
 HOLIDAY))  
 THEN  
 (APPLICANT IS IN FULL-TIME WORK))  
 RULE-43 (IF ((APPLICANT IS AWAY FROM WORK WITHOUT GOOD REASON))  
 THEN  
 (APPLICANT IS IN FULL-TIME WORK))  
 RULE-44 (IF ((APPLICANT IS UNEMPLOYED BUT STILL HAS FINAL EARNINGS  
 FROM LAST JOB))

THEN  
 (APPLICANT IS IN FULL-TIME WORK))  
 RULE-45 (IF ((APPLICANT IS UNEMPLOYED BUT LESS THAN 2 WEEKS AGO WAS  
 SELF-EMPLOYED))  
 THEN  
 (APPLICANT IS IN FULL-TIME WORK))  
 RULE-46 (IF ((APPLICANT ACCORDS WITH RULES CONCERNING 6 MONTHS  
 PRECEDING CLAIM))  
 THEN  
 (APPLICANT CAN MAKE A CLAIM FOR SUPPLEMENTARY BENEFIT))  
 RULE-47 (IF ((DURING 6 MONTHS PRECEDING CLAIM APPLICANT HAS WORKED  
 FOR AT LEAST 8 HOURS PER WEEK))  
 THEN  
 (APPLICANT ACCORDS WITH RULES CONCERNING 6 MONTHS  
 PRECEDING CLAIM))  
 RULE-48 (IF ((DURING 6 MONTHS PRECEDING CLAIM APPLICANT HAS SIGNED ON  
 AS AVAILABLE FOR WORK OR REGISTERED WITH JOB  
 CENTRE OR CAREERS OFFICE))  
 THEN  
 (APPLICANT ACCORDS WITH RULES CONCERNING 6 MONTHS  
 PRECEDING CLAIM))  
 RULE-49 (IF ((DURING 6 MONTHS PRECEDING CLAIM APPLICANT WAS INCAPABLE  
 OF WORK BECAUSE OF ILLNESS OR DISABILITY AND  
 RECEIVED BENEFIT AS A RESULT))  
 THEN  
 (APPLICANT ACCORDS WITH RULES CONCERNING 6 MONTHS  
 PRECEDING CLAIM))  
 RULE-50 (IF ((DURING 6 MONTHS PRECEDING CLAIM APPLICANT WAS OFF WORK  
 DUE TO A TRADE DISPUTE AT THE PLACE HE/SHE WORKS)  
 )  
 THEN  
 (APPLICANT ACCORDS WITH RULES CONCERNING 6 MONTHS  
 PRECEDING CLAIM))

## APPENDIX E

### REFERENCES

- Addler M., [1975] "A Computer Based Welfare Benefits Information System: The Inverclyde Project", IBM(UK) Scientific Centre.
- Allbeson J. & Douglas J., [1984] "National Welfare Benefits Handbook", Child Poverty Action Group, London.
- Ades T., [1981] "Time for a Purge", in Cognition 10, 7 - 15.
- Allen L.E. [1957] "Symbolic Logic: A Razor-Edged Tool for Drafting and Interpreting Legal Documents", Yale Law Journal, Vol. 66, pp 833-879.
- Allen L.E. & Caldwell M.E., [1963] "Modern Logic and Judicial Decision Making: A Sketch of One View" in "Law and Contemporary Problems", Vol XXVIII, No 1.
- Allen L.E., [1974] "Formalizing Hohfeldian analysis to clarify the multiple senses of "Legal Right": A powerful tool for the electronic age", South Calif. Law Review Vol. 48 pp 428-487.
- Allen L.E., [1980] "Language, Law and Logic: Plain Legal Drafting for the Electronic Age", in Niblett [1980].

- Alvey [1982] "A Program for Advanced Information Technology", HMSO, London.
- Austin J. [1863] "Lectures on Jurisprudence", London.
- Berg H.K., Boebert W.E., Franta W.R., Moher T.G., [1982] "Formal Methods of Program Verification and Specification", Prentice-Hall.
- Bing J. & Harvold T., [1977] "Legal Decisions and Information Systems", Universitetsforlaget, Oslo.
- Bloomfield B.P. [forthcoming] "Modelling the World: The Social Constructions of Systems Analysts", Basil Blackwell.
- Bond A.H. (Ed.) [1981] "Machine Intelligence", Infotech State of the Art Report, Series 9, No. 3, Pergamon.
- Boole G., [1854] "An Investigation of the Laws of Thought, on which are founded the Mathematical Theories of Logic and Probabilities", London.
- Brachman R.J., [1983] "What IS-A Is and Isn't: An Analysis of Taxonomic Links in Semantic Networks", in Computer, Vol. 16, No. 10.
- Buchanan B.G., Sutherland G. Feigenbaum E.A. [1969] "HEURISTIC DENDRAL: a program for generating explanatory hypothesis in organic chemistry", in Meltzer B. & Michie D. (eds) Machine Intelligence, Vol. 4, pp 209-254.
- Buchanan B.G. & Headrick T.E., [1970] "Some Speculation About Artificial Intelligence and Legal Reasoning", Stanford Law Review, Vol. 23, 40-62.
- Buchanan B.G., [1979] "Issues of Representation in Conveying the Scope and Limitations of Intelligent Assistant Programs", in Hayes J.E., Michie D., Mikulich L.I. (eds) Machine Intelligence, Vol 9, pp407-426, Ellis Horwood.

- Bundy A., [1983] "The Computer Modelling of Mathematical Reasoning", Academic Press.
- Campbell J.A. [1982] "The Expert Computer and Professional Negligence: Who is Liable?", Research Report R-109, University of Exeter.
- Catt I. [1973] "Computer Worship", Pitmans.
- Chomsky N., [1957] "Syntactic Structures", Mouton.
- Clancey W.J., [1979] "Transfer of Rule Based Expertise Through a Tutorial Dialogue", PhD Dissertation, Stanford University.
- Clancey W.J. & Letsinger R., [1981] "NEOMYCIN: Reconfiguring a rule-based expert system for application to teaching", in Int. Joint conference on Artificial Intelligence, pp829-836.
- Clark K.L., [1978] "Negation as Failure" in "Logic and Data Bases", H. Gallaire & J. Minker (Eds), Plenum Press, New York.
- Cook S. & Stamper R., [1980] "LEGOL as a Tool for the Study of Bureacracy" in Lucas [1980].
- Council of Europe [1982] "Artificial Intelligece and Lingusitic Problems in Legal Data Processing Systems", Strasbourg.
- Cory H.T., Hammond P., Kowalski R., Kriwaczek F., Sadri F., Sergot M., [1984] "The British Nationality Act as a Logic Program", Department of Computing, Imperial College, London.
- Craig P.P, [1983] "Administrative Law", Sweet & Maxwell, London.
- Davis R. & King J., [1976] "An Overview of Production Systems" In Machine Intelligence 8, Elcock E.W. & Michie D. (Eds.).

- Davis R., [1977a] "Interactive Transfer of Expertise: Acquisition of New Inference Rules", in Int. Joint. Conf. on Artificial Intelligence, pp321-328.
- Davis R. & Buchanan B.G., [1977b] "Production rules as a representation for a knowledge-based consultation program" in Artificial Intelligence, Vol. 8, pp15-45.
- Davis R., [1980a] "Content Reasoning: Reasoning about Rules", in Artificial Intelligence, Vol. 15, No. 3.
- Davis R., [1980b] "Meta-Rules: Reasoning about Control", in Artificial Intelligence, Vol 15, No. 3.
- Davis R., [1982] "Expert Systems: where are we are where do we go from here?", AI Lab Memo 665, MIT, June.
- De Long H., [1970] "A Profile of Mathematical Logic", Addison-Wesley.
- De Smith S.A., [1980] "Judicial Review of Administrative Action", Stevens & Sons, London. 4th Edition.
- Del Vecchio G., [1914] "Formal Bases of Law", in Modern Legal Philosophy, Series X, Boston.
- Dillman I. [1973] "Induction and Deduction: A Study in Wittgenstein", Basil Blackwell.
- Dreyfus H.L., [1981] "From Micro-Worlds to Knowledge Representation" in Haugeland [1981].
- du Feu D, [1980] "Selecting Welfare Benefits by Computer," in Niblett [1980].
- Duda R.O., Gaschnig J., Hart P., [1979] "Model Design in the Prospector Consultant System for Mineral Exploration", in Michie [1979].



- Duda R.O., & Shortliffe E.H., [1983] "Expert Systems Research" in Science 220, pp261-268.
- Durkheim E. [1982] "The Rules of Sociological Method", Macmillan.
- Edwards W., [1972] "N=1: Diagnosis in Unique Cases", in Jacquez [1972], pp 139-151.
- Encyclopedia of Philosophy, [1967] Collier-MacMillan.
- Feigenbaum E., [1971] "On generality and problem solving: A case study using DENDRAL programs", in Machine Intelligence 6, Meltzer B. & Michie D. (eds), pp 165-190, Ellis Horwood.
- Feigenbaum E.A., [1977] "The Art of Artificial Intelligence: Themes and case studies of knowledge engineering", in Int. Joint Conference on Artificial Intelligence.
- Feigenbaum E., [1980] "Expert Systems in the 1980's" in Proceedings, Day 3, State of the Art Review, Infotech Ltd.
- Fleck L., [1979] "Genesis and Development of a Scientific Fact", Chicago.
- Fraunce A., [1588] "The Lawyer's Logicke", fascimile printed by the Scolar Press.
- Forgy C.L., [1979] "On the Efficient Implementation of Production Systems", PhD Thesis, Carnegie-Mellon University.
- Foster A. [1984] "Beauty and the Beast", in Campbell C.M. (ed), "Legal Data Processing", Sweet & Maxwell.
- Foundos S., [1971] "Cybernetics and Tax Law, Predictions of Courts Tax Cases", Association of Certified Accountants, London.

- Friedland P., [1981] "Acquisition of procedural knowledge from domain experts", Int. Joint Conf. on Artificial Intelligence, pp856-861.
- Georgeff M.P., [1982] "Procedural Control in Production Systems" in Artificial Intelligence 18, pp175-201.
- Godel K., [1930] "Die Vollständigkeit der Axiome des logischen Funktionenkalküls", Monatshefte für Mathematik und Physik, Vol. 37, pp 349-360.
- Godel K., [1931] "Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I", Ibid., Vol. 38, pp 173-198.
- Goldstein I.P. & Grimson E., [1977] "Annotated Production Systems, A Model for Skill Acquisition", in Int. Joint Conference on Artificial Intelligence, pp 311/316.
- Griffith J.A., [1977] "The Politics of the Judiciary", Humanities Press (also 3rd Edition [1985]).
- Guest A.G., [1961] "Logic in the Law", in "Oxford Essays in Jurisprudence" (First series), Oxford.
- Haack S., [1978] "Philosophy of Logics", Cambridge.
- Haines C.G. [1922] "General Observations on the Effects of Personal, Political and Economic Influences in the Decisions of Judges", 17 Illinois Law Review, 96.
- Hammond P., [1983a] "Representation of DHSS Regulations as a Logic Program", Report No. 82/26, September, Dept. of Computing, Imperial College, London.
- Hammond P., [1983b] "A Listing of a Prolog Program Describing Entitlement to Supplementary Benefit", June, Dept. of Computing, Imperial College, London.

- Hammond P. & Sergot M., [1983c] "A PROLOG Shell for Logic Based Expert Systems", Proc. 3rd BCS Experts Systems Conf., pp 95-104.
- Hanfling O., [1981] "Logical Positivism", Basil Blackwell, Oxford.
- Hart H.L.A., [1961] "The Concept of Law", Oxford.
- Hart H.L.A. [1973] "Bentham on Legal Rights", in Simpson A.W.B. (ed) "Oxford Essays in Jurisprudence", Second Series, Oxford.
- Hart P.E., [1975] "Progress on a Computer Based Consultant", in Int. Joint Conference on Artificial Intelligence, pp 831-841.
- Hellawell R., [1980] "A Computer Program for Legal Planning and Analysis: Taxation of Stock Redemptions", in 80 Columbia Law Review 1363.
- Howell W.S., [1956] "Logic & Rhetoric in England 1500-1700", Russell & Russell.
- Hunter G., [1971] "Metalogic", MacMillan, London.
- Jacques J.A. [1972] (ed) "Computer Diagnosis and Diagnostic Methods", Proc. of the 2nd Conf. on the Diagnostic Process, Ann Arbor, Michigan.
- Jensen O.C., [1957] "The Nature of Legal Argument", Blackwell, Oxford.
- Jensen K. & Wirth N., [1975] "Pascal User Manual and Report", Springer, New York.
- Kelsen H., [1967] (Tr. Max Knight) "Pure Theory of Law", Univ. of California Press.
- King J, [1976] "Analysis and KRL Implementation of a Current Legal Reasoning Program Design", mimeograph, Stanford University.

- Kilminster C.W., [1967] "Language, Logic and Mathematics", London.
- Kline M., [1972] "Mathematics in Western Culture", Pelican, London.
- Kline M., [1980] "Mathematics: The Loss of Certainty", Oxford.
- Kondos G.S., [1971] "JURIS: Remote Terminal Legal Information Retrieval at the U.S. Department of Justice", Law and Computer Technology, 147-155.
- Kowalski R., [1979] "Logic for Problem Solving", North Holland.
- Kowalski R., [1980] Quoted in Special Issue on Knowledge Representation, SIGART Newsletter, No. 70.
- Lakatos I., [1976] "Proofs and Refutations, The Logic of Scientific Discovery", CUP.
- Lapham N. [1985] Personal Communication, Alvey/DHSS Demonstrator Project.
- Larson S.E. & Williams M.E. [1980] "Computer Assisted Legal Research" in Annual Review of Information Science and Technology, Vol. 15.
- Lawlor R.C., [1980] "Computer analysis of judicial decisions", in Niblett [1980].
- Lewis C.I., [1918] "A Survey of Symbolic Logic", Univ of California Press, Berkeley.
- Levi E.H. [1949] "An Introduction to Legal Reasoning", Chicago.
- Levi-Strauss C., [1966] "The Savage Mind", Weidenfeld & Nicolson, London.
- Lighthill Sir J., [1973] "Artificial Intelligence: a paper symposium", Science Research Council, London.

- Lindsay R.K., Buchanan B.G., Feigenbaum E.A., Lederberg J., [1980]  
"Applications of artificial intelligence for organic chemistry.  
The DENDRAL project." McGraw-Hill.
- L.T. [1945] Parker v London and North Eastern Railway Company, L.T.  
176, 137(1945).
- Lucas H.C., Land F.F., Lincoln T.J.L., Supper K,(Eds.) [1980] "The  
Information Systems Environment", North Holland.
- MacCormick D.N., [1978] "Legal Reasoning and Legal Theory", Oxford Univ.  
Press.
- MacCormick D.N., [1982] "The nature of legal reasoning: A brief reply  
to Dr Wilson" in Legal Studies Vol 2 No 3.
- McCarty L.T., [1977] "Reflections on TAXMAN: An Experiment in  
Artificial Intelligence and Legal Reasoning", in Harvard Law  
Review, Vol. 90, No. 5.
- McCarty L.T., [1980a] "The TAXMAN Project: Towards a cognitive theory  
of legal argument" in Niblett [1980].
- McCarty L.T., [1980b] "Some Requirements for a Computer-Based Legal  
Consultant", Report LRP-TR-8, Laboratory for Computer Science  
Research, Rutgers University.
- MacDonald [1983] "Immigration Law and Practice in the United Kingdom",  
Butterworths.
- McDermott D., [1981] "Artificial Intelligence Meets Natural Stupidity"  
in Haugeland [1981].
- McGettrick A.D. [1980] "The Definition of Programming Languages",  
Cambridge.

McGettrick A.D. [1982] "Program Verification Using ADA", Cambridge.

Macklovitch E., [1984] "Recent Canadian Experience in Machine Translation" in Proc. Int. Conf. on Machine Translation, Cranfield.

Malik R. [1975] "And Tomorrow The World? Inside IBM", Millington.

Martin J. & McClure C., [1983] "Software Maintenance, The Problem and its Solution", Prentice-Hall.

Mason P. & Jones S., [1980] "Programming the Law" in Lucas [1980].

Mates B. [1961] "Stoic Logic", University of California Press.

Mehl L., [1958] "Automation in the legal world", in Symposium on Mechanization of Thought Processes, National Physical Laboratory, HMSO.

Meldman J.A., [1975] "A Preliminary Study in Computer-Aided Legal Analysis", PhD Thesis, MIT.

Michie D., [1979] "Expert Systems in the Micro-electronic Age", Edinburgh Univ. Press.

Milne R. & Strachey C., [1976] "A Theory of Programming Language Semantics", London.

Minsky M., [1968] "Semantic Information Processing", MIT.

Minsky M & Papert S., [1970] Draft of a proposal to ARPA for research on artificial intelligence at MIT, 1970-71.

Minsky M., [1975] "A framework for representing knowledge" in Winston [1975].

- Moles R.N. [1985] "Rule and Definition in Jurisprudence: A Critique of H.L.A. Hart's response to John Austin", unpublished PhD Thesis, Edinburgh University.
- Morrise M., [1980] "Emerging Computer Assisted Legal Advisory Systems", in Brigham Young University Law Review, Vol. 116.
- Mulder R.V., Oskamp A., van der Heyden W., Gubby H.M. [1982] "Sentencing by Computer: an experiment", Norwegian Research Centre for Computers and Law, Oslo University.
- Nagel E. & Newman J.R., [1971] "Godel's Proof", Routledge, London.
- Nagel S.S. & Neef M., [1977] "The Legal Process; Modeling the System", Sage.
- Newell A. & Simon H., [1972] "Human Problem Solving", Prentice Hall.
- Niblett B., [1980] "Computer Science and Law", Cambridge.
- Niblett B., [1981] "Expert Systems for Lawyers", in Computers and Law, No 28.
- Nickerson R.S., [1981] "Why interactive Computer Systems are sometimes not used by people who might benefit from them", in Inter. J. Man-Machine Studies 15, pp496-483.
- Ong W.J., [1958] "Ramus, Method and the Decay of Dialogue", Harvard University Press.
- Oddy R.N., S.E. Robertson, von Rijsbergen C.J. [1981] "Information Retrieval Research", Butterworths. Octagon Books, New York.
- O'Shea T & Young R., [1978] "A Production Rule Account of Children's Errors on Subtraction", in Proc. AISB Conf.

- Patterson A. [1982] "The Law Lords", Macmillan.
- Popp W.G. & Schlink B., [1975] "Judith, A Computer Program to Advise Lawyers in Reasoning a Case." in Jurimetrics Journal, Summer, 1975.
- Post E.L., [1943] "Formal reductions of the general combinatorial decision problem" in Am. J. Math. 56 pp 197-268.
- Quillian M.R., [1968] "Semantic Memory", in Minsky [1968], pp 354-402.
- Raphael B., [1976] "The Thinking Computer: Mind Inside Matter", W.H. Freeman & Co, San Francisco.
- Reboh R., [1981] "Knowledge Engineering Techniques and Tools for Expert Systems", Dissertation No. 71, Software Systems Research Centre, Linkoping University, Sweden.
- Renton Sir D., [1975] "The Renton Report: Committee on the Preparation of Legislation", HMSO, London.
- Ritchie G.D. & Hanna F.K. [1984] "AM: A Case Study in AI Methodology", in Artificial Intelligence, Vol. 23, No. 3.
- Robson P. & Watchman P. [1981] "Justice, Lord Denning and the Constitution", Gower.
- Rogers W.V.H., [1979] "Winfield & Jolowicz on Tort", 11th Edition, Sweet and Maxwell, London.
- Rooney B.J., [1984] "Dynamics of the Expert Systems Community", paper presented to ECAI, Piza, Sept. 1984.
- Rubin J.S., [1976] "Computer-Assisted Legal Research: LEXIS", The Harvard University Program on Information Technologies and Public Policy, 8th March.



- Russell B., [1903] "The Principles of Mathematics", George Allen & Unwin.
- Russell B., [1956] "Portraits from Memory and Other Essays", George Allen & Unwin.
- Russell B., [1959] "My Philosophical Development", George Allen & Unwin.
- Russell B., [1980] "An Inquiry into Meaning and Truth," George Allen & Unwin, London.
- Schubert G., [1963] "Judicial Attitudes and Voting Behaviour: The 1961 Term of the United States Supreme Court", in "Jurimetrics", Baade H.W. (ed), Basic Books Inc., London (also published in "Law and Contemporary Problems", Vol 28, No 1, Winter 1963.)
- Scott A.C., Clancey W.J., Davis R., Shortliffe E.H. [1977] "Explanation Capabilities of Production-Based Consultation Systems", American Journal of Computational Linguistics, Microfiche 62.
- Schank R.C., Kolodner J.L., DeJong G., [1981] "Conceptual Information Retrieval", in Oddy [1981].
- Sergot M., [1980] "LEGOL as a Logic Programming Language", LEGOL Bulletin No. 9, May 1980.
- Sergot M., [1982] "Prospects for Representing the Law as Logic Programs" in "Logic Programming", K.L. Clark & S.A. Tarnlund (Eds), Academic Press.
- Simon H.A. "Artificial Intelligence Systems that Understand", in International Joint Conference on Artificial Intelligence, pp 1059-1073.

- Sizer T.R.H. & Kelman A. (Ed.) [1982] "Computer Generated Output as Admissible Evidence in Civil and Criminal Cases", Heyden & Son.
- Slayton P., [1974] "Electronic Legal Retrieval", Report prepared for the Department of Communications of the Government of Canada, Ottawa.
- Shortliffe E.H., [1975] "Mycin: A rule-based computer program for advising physicians regarding antimicrobial therapy selection.", Stanford Univ. PhD Thesis.
- Shortliffe E.H., [1982] "Barriers there, but can be overcome", in IEEE Engineering in Medicine and Biology Magazine, Vol. 1, Pt. 32, pp 16-18.
- Spengler J.J., [1963] "Machine-Made Justice: Some Implications", Law and Contemporary Problems, Vol XXVII, Winter, No. 1.
- Sprowl J., [1980] "Automated Assembly of Legal Documents", in Niblett [1980].
- Stamper R., [1977] "Physical Objects, Human Discourse and Formal Systems" in Nijssen G.M. (Ed.) "Architecture and Models in Database Management Systems", North Holland.
- Stamper R., [1980a] "Text Plus Meaning: A Goal for Law Retrieval in the Eighties" in "Lawyers in the Eighties", Society for Computers and Law.
- Stamper R. Tagg C. Mason P. Cook S. Marks J., [1980b] "Developing the LEGOL Semantic Grammar", LEGOL Paper L48, LSE, London.
- Stamper R., [1980c] "LEGOL: Modelling legal rules by Computer" in Niblett [1980].

- Stamper R., [1980d] "Computer-aided legal reasoning", Poly Law Review, Spring Vol. 5, pp41-52.
- Stefik M., Aikins J., Balzer r., Benoit J., Brinbaum L., Hayes-Roth F., Sacerdoti E. [1982] "The Organization of Expert Systems, A Tutorial", in Artificial Intelligence, Vol 18, pp 175-202.
- Stone J., [1964] "Legal System and Lawyer's Reasonings", Stanford Univ. Press.
- Styazhkin N.I., [1969] "History of Mathematical Logic from Leibniz to Peano", MIT Press.
- Summers R.S., [1961] "A Note on Symbolic Logic and the Law", in Journal of Legal Education, Vol. 13, pp 486-492.
- Swartout W.R., [1983] "XPLAIN: a system for Creating and Explaining Expert Consultating Programs", AI 21, pp 285-325.
- Tammelo I., [1969] "Outlines of Modern Legal Logic", Franz Steiner Verlag, Weisbaden.
- Tapper C., [1973] "Computers and Law", London.
- Teach R.L. & Shortliffe E.H. [1981] "Physician Attitues and Consultation Systems", in Computers and Biomedical Research 14, pp 542-558.
- Twining W. & Miers D., [1982] "How to Do Things With Rules", (2nd Ed.), Weidenfeld & Nicholson, London.
- Van Melle W., Shortliffe E.H., Buchanan B.G. [1981] "EMYCIN: A domain independant system that aids in constructing knowledge-based consultation programs", in Bond [1981].

- Wang H., [1954] "The Formalization of Mathematics" in The Journal of Symbolic Logic, Vol. 19, No. 4.
- Wasserstrom R.A., [1961] "The Judicial Decision", Stanford University Press.
- Waterman D.A. & Hayes-Roth F., [1978a] "Pattern-Directed Inference Systems", Academic Press.
- Waterman D.A., & Hayes-Roth F., [1978b] "An Overview of Pattern-Directed Inference Systems", in Waterman [1978a].
- Waterman D.A. & Peterson M.A., [1981] "Models of Legal Decisionmaking", Rand Corporation, Santa Monica.
- White P.D., [1980] "Philosophy and Law: Some observations on MacCormick's Legal Reasoning and Legal Theory", Michigan Law Review Vol. 78, p 737-742.
- Whitehead A.N. & Russell B. [1910/13] "Principia Mathematica", Cambridge.
- Wilson A., [1982] "The nature of legal argument: A commentary with special reference to Professor MacCormick's theory", in Legal Studies Vol. 2 No. 3.
- Winston P.H., [1975] "The Psychology of Computer Vision", McGraw Hill.
- Wittgenstein L., [1956] "Remarks on the Foundations of Mathematics", (tr. G.E.M. Anscombe), Basil Blackwell.
- Wittgenstein L., [1958] "Philosophical Investigations", (tr. G.E.M. Anscombe), Basil Blackwell.
- Wittgenstein L., [1969] "The Blue and Brown Books", Basil Blackwell.

Wittgenstein L., [1981] "Tractatus Logico-Philosophicus", (tr. C.K. Ogden), Routledge Kegan Paul, London.

Woods W.A., [1975] "What's in a link: foundations for semantic networks." in "Representations and understanding: Studies in Cognitive Science", Bobrow D.G. & Collins A (eds), Academic Press, New York.

Woolgar S., [1984] "Why Not a Sociology of Machines? ... ", Dept. of Sociology, Brunel University, England.